

**AÇIKLAMALI
UYGULAMA ÖRNEKLİ
BİLGİSAYAR**

**BİLGİSAYARA GİRİŞ
MS-DOS
PCTOOLS
PW
BASIC**

**GÜVEN TANIŞ
Öğretim Görevlisi**

Copyright © Güven Tanış

Bu kitabın tüm hakkı saklıdır. Hiç bir kimse ve kuruluş, yurt içinde ya da yurt dışında kaynak göstermeksizin, kitaptan bir bölüm yayınlamayabilir, başka bir dile çeviremez ve herhangi bir yöntemle (Disket, fotokopi vb.) çoğaltamaz ya da manyetik ortamda saklayamaz.

Aksi davranışlar hakkında telif yasası hükümleri uygulanır.

Dizgi, Editör, Çizimler, Kapak: Güven TANIŞ

Ağustos 1994

ISBN 975-95337-2-3

E-Kitap Notu:

Windows işletim sistemi çıktığından günümüzde artık kullanılmasa da birilerinin işine yarayacağını düşünerek, hazırlamış olduğum bu kitabı e-kitap olarak sunuyorum. Dağıtımını serbesttir. Bir not düşmek istiyorum ne kadar artık dos kullanılmasa da Windows'un halen sorunlu olduğu durumlarda ben yararlanıyorum. Bir gün bakarsınız sizin de işinize yarar.

**Güven Tanış
Öğretim Görevlisi**

ÖNSÖZ

Annem Kamuran ve Babam Fikret'e...

Değerli meslektaşlarım, sevgili öğrenciler ve bilgisayar kullanıcıları. Bildiğimiz gibi bilgisayarlar günümüzde bütün yaşamımızı etkilemektedir. Artık bilgisayar öğrenmek bir zorunluluk olmuştur. Kitabın adını görünce biraz düşünmüşsünüzdür. Açıklamalı ve Uygulama Örneklili bir kitap nasıl olur? Kitapta gerçekten anlaşılabilir açıklamalar ve ekran gösterimleri ile uygulama örnekleri verilmiştir.

Bu kitabı yıllardır verdiğim bilgisayar derslerinde öğrencilerin ve öğretmenlerin karşılaştıkları sorunları da gözönüne alarak, eğitimci bir anlayışla hazırladım. Sizlere yararlı olacağını düşünüyorum.

2. ve 3. bölümde bilgisayarın tanıtımı, birimleri ve çalışma sistemi verilerek, bilgisayarın mantığının kavranması amaçlanmıştır.

4. bölümde işletim sistemi ve günümüzde en çok kullanılan MS-DOS işletim sisteminin çalışma mantığı ve komutları verilmiştir.

5. bölüm bellek hizmet programlarını kapsamaktadır. Bu bölümde yine en çok kullanılan ve pratik olan PCTOOLS verilmiştir.

6. bölümde Belge Düzenleyici (kelime işlem) programlarının mantığı ve PROFESSIONAL WRITE (PW) verilmiştir.

7. bölümde sayı sistemleri tanıtılmış. Sayı sistemlerinde dört işlem ve geçişler verilmiştir.

8. bölüm programlamanın mantığını vermektedir. Kitapta BASIC programlama dili verilmekle kalmamış, programlamanın genel mantığı verilerek diğer dillere geçiş kolaylaştırılmıştır.

9. 10. ve 11. bölümlerde BASIC programlama dilinin çalışma sistemi ve komutları verilirken yine amaç, BASIC'I öğretmekle birlikte programlama mantığını vermektir.

12. bölümde ek bilgiler verilmiştir. Bunlar MS-DOS, BASIC hata mesajları, ASCII ve HEX karakter kodları listesi. Bilgi İşlem Terimleri Sözlüğü'dür.

Sizlere yardımcı olacağını düşünerek, çalışmalarınızda başarılar diler, saygı ve sevgiler sunarım.

**Güven Tanış
Öğretim Görevlisi**

ÖZGEÇMİŞ

Güven Tanış 1961'de Konya – Ereğli'de doğdu. İlk ve Orta öğrenimini Ereğli'de yaptı. Yüksek öğrenimini Gazi Üniversitesi - Mesleki Eğitim Fakültesi - Ticaret ve Turizm Eğitimi - Büro Yönetimi Ana Bilim Dalı'nda tamamladı.

1984 yılında Meslek Dersleri Öğretmeni olarak Çanakkale Biga Ticaret Meslek Lisesi'nde göreve başladı. 1991 yılında Bayburt Ticaret Meslek Lisesi'ne tayin oldu. 1993 yılından bu yana Akdeniz Üniversitesi - Sosyal Bilimler Meslek Yüksekokulu'nda Öğretim Görevlisi olarak görev yapmaktadır.

Yapıtları: Çeşitli sanat ve edebiyat dergilerinde şiir ve yazıları, 1992 yılında ilk kitabı olan Mikrobilgisayarlar İçin - MS-DOS ve BASIC yayınlandı. 1993 yılında Anadolu Ticaret ve Ticaret Meslek Liseleri için yazdığı BİLGİSAYAR 1 - 2, Talim ve Terbiye Kurulu'nca ders kitabı olarak kabul edildi.

İLETİŞİM
e-posta: gtanis@gmail.com
Güven Tanış
P.K. 479
07003 ANTALYA

ISBN 975 - 95337 - 2 - 3

ANTALYA

İÇİNDEKİLER

ÖZGEÇMİŞ	4
BÖLÜM I.....	21
BİLGİSAYARA GİRİŞ	21
A. BİLGİSAYARIN TANIMI	21
B. BİLGİSAYARIN TARİHSEL GELİŞİMİ	21
1. BİRİNCİ KUŞAK BİLGİSAYARLAR. (1945-1956)	22
2. İKİNCİ KUŞAK BİLGİSAYARLAR. (1957-1964).....	22
3. ÜÇÜNCÜ KUŞAK BİLGİSAYARLAR. (1965-1970)	23
4. DÖRDÜNCÜ ve BEŞİNCİ KUŞAK BİLGİSAYARLAR.....	23
(1970 ve Sonrası).....	23
C. BİLGİSAYARLARIN KULLANIM ALANLARI	23
D. BİLGİSAYAR TÜRLERİ.....	24
1. Çalışmalarına Göre Bilgisayarlar:	24
a. Sayma Tekniğine Dayalı Bilgisayarlar:	24
b. Ölçme Tekniğine Dayalı Bilgisayarlar:.....	24
c. Karma Tip Bilgisayarlar:.....	25
2. Büyüklüklerine Göre Bilgisayarlar.....	25
E. BİLGİSAYAR KAVRAMLARI	25
1. BIT, BYTE, K.BYTE, M.BYTE, G.BYTE, T.BYTE	25
2. DONANIM (HARDWARE).....	26
3. YAZILIM (SOFTWARE).....	28
4. İŞLETİM SİSTEMİ	28
BÖLÜM II	30
BİLGİSAYARIN GENEL YAPISI.....	30
A. GİRİŞ BİRİMİ	30
B. MERKEZİ İŞLEM BİRİMİ (M.İ.B.)	30
1. ARİTMETİK ve MANTIK BİRİMİ	30
2. KONTROL (DENETİM) BİRİMİ	30
3. BELLEK BİRİMİ	30
C. ÇIKIŞ BİRİMİ.....	33
D. YARDIMCI BELLEK BİRİMİ.....	33
E. GİRİŞ VE ÇIKIŞ BİRİMLERİ.....	33
1. EKРАН VE EKРАН ÇEŞİTLERİ	33
2. KLAVYE ve KLAVYE ÇEŞİTLERİ	35
3. YAZICI ve YAZICI ÇEŞİTLERİ	36
4. DİSKET ve DİSKET ÇEŞİTLERİ.....	38
c. Sabit Disk.....	41
5. DRIVER ve DRIVER ÇEŞİTLERİ	42
a. 3.5 inch'lik sürücü	42

b. 5.25 İnce'lik sürücü	42
c. CD Sürücü	43
6. BİLGİSAYARDA KULLANILAN DİĞER YAN BİRİMLER	43
a. Mouse (Fare)	43
b. Scanner (Tarayıcı).....	43
c. Modem	44
d. Plotter (Çizici).....	45
e. Oyun Çubuğu (Joystick).....	45
f. Işık Kalem (Light Pen).....	45
g. Teyp	45
7. BİLGİSAYARLAR ARASINDA BAĞLANTI	45
BÖLÜM III	47
BİLGİSAYARIN ÇALIŞMA İLKESİ	47
A. DİSKETİN SÜRÜCÜYE TAKILMASI	47
B. SİSTEMİN AÇILMASI	47
BÖLÜM IV	50
İŞLETİM SİSTEMİ	50
A. GİRİŞ	50
B. İŞLETİM SİSTEMİ ÇEŞİTLERİ	50
C. DOS İŞLETİM SİSTEMİNE GİRİŞ	51
D. DOS İŞLETİM SİSTEMİ KAVRAMLARI	51
1. SÜRÜCÜ ADI	51
2. YOL ADI (PATH NAME).....	51
3. DİZİN (DIRECTORY)	52
a. ANA DİZİN (ROOT DIRECTORY)	52
b. ALT DİZİN (SUB DIRECTORY)	52
4. DOSYA (KÜTÜK, FILE)	54
a. DOSYA ADI VE KURALLARI	56
b. DOSYA UZANTISI.....	57
5. DOSYA ADLARINDA KULLANILAN JOKER.....	57
KARAKTERLER	57
6. İÇ VE DIŞ KOMUTLAR	58
E. İŞLETİM SİSTEMİNDE KLAVYE KULLANIMI	59
F. MS-DOS KOMUTLARI	69
1. DİSKİN HAZIRLANMASI	70
2. TARİHİ GÖSTERME - AYARLAMA.....	75
3. ZAMANI GÖSTERME - AYARLAMA	76
4. DİSK ETİKETİ YAZMA	77
5. MS-DOS UYARLAMA (VERSİYON) NUMARASI	77
6. DİSK ETİKETİNİ GÖRME	78
7. SİSTEM AKTARMA	78
8. DOSYA ve DİZİN LİSTESİNİ GÖRME.....	79

9. EKRANI SİLME.....	85
10. DİZİN AÇMA, DEĞİŞTİRME ve SİLME	85
a. DİZİN AÇMA	85
b. DİZİN DEĞİŞTİRME.....	86
c. DİZİN ADI SİLME	87
11. DOSYA KOPYALAMA.....	88
12. DOSYA TAŞIMA ve DİZİN ADI DEĞİŞTİRME	93
13. AYNI İSİMDEKİ DOSYALARI KOPYALAMA	94
14. DİSKET KOPYALAMA	94
15. DOSYA SİLME	96
16. DİZİN SİLME.....	97
17. DOSYA İÇERİĞİNİ GÖRME.....	98
18. DOSYA ADI DEĞİŞTİRME.....	99
19. DİSK KONTROLÜ	99
20. DİSK ÇÖZÜMLEMESİ ve ONARIM.....	101
21. KOMUT SATIRINI DÜZENLEME.....	102
22. YEDEKLEME	104
23. YEDEKLENMİŞ DOSYALARI GERİ KOPYALAMA	105
24. YEDEKLEME ve GERİ YÜKLEME.....	106
25. DOSYA NİTELİĞİNİ DEĞİŞTİRME.....	109
26. DOSYA KARŞILAŞTIRMA.....	109
27. DİSKET KARŞILAŞTIRMA	110
28. VİRÜS İŞLEMLERİ.....	111
a. VİRÜS KONTROLÜ	111
b. VİRÜS BULMA, TEMİZLEME	112
29. SİLİNİMİŞ DOSYALARI KURTARMA	113
30. FORMATI GERİ ALMA.....	115
31. ARAMA YOLU BELİRTME	116
32. KOMUT UYARISI.....	116
33. BELLEK GÖSTERME.....	117
34. BELLEK AYARLAMA.....	118
35. PROGRAMLARI KONVANSİYONEL BELLEĞİN ÜST	120
KISMINA ATMA.....	120
36. PROGRAMLARI ÜST BELLEĞE ATMA	120
37. DİSK KAPASİTESİNİ ARTTIRMA.....	121
38. DİSK PARÇALANMASINI GİDERME.....	122
39. HARD DİSKİ DÜZENLEME	124
40. DİSK CACHE PROGRAMI.....	126
41. UYARLAMA TABLOSUNU AYARLAMA	127
42. VERİ ARAMA.....	128
43. KLAVYE DEĞİŞTİRME	128
44. DOSYA SIRALAMA	130

45. GENİŞLETİLMİŞ KARAKTER SETİ.....	131
46. YAZICIYA GRAFİK SETİNİ YÜKLEME.....	132
47. MOD DEĞİŞTİRME	132
48. MS-DOS EDITÖRÜ	134
49. SIKIŞTIRILMIŞ DOSYALARI AÇMA	139
50. SİSTEM KONTROLU	139
51. DEBUG.....	141
G. TOPLU İŞLEM DOSYALARI (BATCH) ve	142
KOMUTLARI.....	142
1. AUTOEXEC.BAT (AÇILIŞ TOPLU İŞLEM DOSYASI).....	143
2. BATCH DOSYASI ALT KOMUTLARI	144
ECHO.....	144
@.....	144
FOR..IN..DO.....	145
GOTO	145
IF	145
CALL	145
CHOICE.....	146
PAUSE.....	147
REM.....	147
SHIFT	147
H. KONFIGÜRASYON DOSYASI	148
(DONANIM ÖZELLİKLERİNİ BELİRLEME).....	148
1. KONFIGÜRASYON DOSYASI KOMUTLARI.....	148
2. KURULABİLİR BİRİM SÜRÜCÜLERİ	152
BÖLÜM V	158
BELLEK HİZMET PROGRAMI	158
A. GİRİŞ	158
B. BELLEK HİZMET PROGRAMLARININ.....	158
TEMEL MANTIĞI.....	158
C. BELLEK HİZMET PROGRAMLARININ.....	158
TEMEL KAVRAMLARI	158
D. PCTOOLS BELLEK HİZMET PROGRAMINA	159
GİRİŞ VE TEMEL AÇIKLAMALAR	159
E. PCTOOLS DOSYA İŞLEMLERİ MENÜSÜNÜN.....	160
AMACI ve KOMUTLARI.....	160
1. DOSYA İŞARETLEME	162
ENTER Tuşu.....	162
2. İŞARETLEMEDEN VAZGEÇME.....	162
F1=UNselect	162
3. DOSYA HAKKINDA DETAYLI BİLGİ.....	163
F2=alt dir lst.....	163

4. DISK İŞLEMLERİ MENÜSÜNE GEÇME.....	163
F3 other menu	163
5. PCTOOLS'TAN ÇIKMA	163
Esc=exit	163
6. AYNI ÖZELLİKTEKİ DOSYALARI GÖRME.....	163
F8=directory LIST argument	163
7. AYNI ÖZELLİKTEKİ DOSYALARI İŞARETLETME.....	164
F9=file SELECTION argument	164
8. SÜRÜCÜ YA DA DİZİN DEĞİŞTİRME	164
F10=chg drive/path	164
9. DOSYA KOPYALAMA.....	165
Copy.....	165
10. DOSYA TAŞIMA.....	167
Move	167
11. DOSYA KARŞILAŞTARMA	168
cOmp.....	168
12. KARAKTER GRUBU ARATMAK ve DEĞİŞTİRMEK	168
Find	168
13. DOSYA ADI DEĞİŞTİRMEK.....	171
Rename	171
14. DOSYA SİLME.....	171
Delete	171
15. DOSYA KONTROLÜ ve DÜZELTME.....	172
Ver	172
16. DOSYALARIN İÇERİĞİNİ GÖRME ve DÜZELTME.....	173
wiew/Edit	173
17. DOSYALARIN NİTELİĞİNİ GÖRME ve DEĞİŞTİRME.....	174
Attrib.....	174
18. BELGE DÜZENLEYİCİ (KELİME İŞLEM, EDITOR) KULLANIMI... 176	
Wordp (WORDPROCESSOR).....	176
19. YAZICIYA YAZDIRMA	180
Print.....	180
20. DOSYA ÖZELLİKLERİNİ YAZICIYA YAZDIRMA	183
List	183
21. DOSYALARI SIRALAMA	184
Sort.....	184
22. YARDIM ALMA.....	185
Help.....	185
F. DİSK İŞLEMLERİ MENÜSÜNÜN AMACI ve	186
KOMUTLARI.....	186
G. DİSK SERVİS KOMUTLARI	187
1. DOSYA İŞLEMLERİ MENÜSÜNE GEÇME	187

F3=file srvc	187
2. SÜRÜCÜ DEĞİŞTİRME	187
F10=F3+chg drive.....	187
3. DİSKET KOPYALAMA	187
Copy.....	187
4. DİSKET KARŞILAŞTIRMA	189
cOmpare.....	189
5. DİSKTE KARAKTER GRUBU ARAMA	191
Find	191
6. DİSK ETİKETİ DEĞİŞTİRME.....	191
Rename	191
7. SEKTÖR TARAMA.....	192
Verify.....	192
8. DİSKİN İÇERİĞİNİ GÖRÜNTÜLEME	193
view/Edit.....	193
9. DİSKİN YAPISAL DURUMUNU GÖRME.....	194
Map.....	194
10. DOSYALARIN ÇALIŞMA DİZİNLERİNİ GÖRME	196
Locate.....	196
11. DİSKET FORMATLAMA	197
iNitalize.....	197
12. DİZİN İŞLEMLERİ	199
Directory maint	199
13. SİLİNMIŞ DOSYA VE DİZİNİ KURTARMA.....	201
Undelete	201
14. SİSTEM BİLGİSİ	203
Info.....	203
15. HARD DİSKİ PARK ETME	205
Park.....	205
16. YARDIM ALMA.....	205
Help.....	205
BÖLÜM VI.....	207
BELE DÜZENLEYİCİ (KELİME İŞLEM, EDİTÖR) PROGRAMI	207
A. GİRİŞ.....	207
B. EDİTÖR PROGRAMLAMANIN TEMEL.....	207
MANTIĞINI KAVRAMA.....	207
C. EDİTÖR PROGRAM ÇEŞİTLERİ.....	207
D. BELGE DÜZENLEYİCİ (EDITOR) PROGRAM KAVRAMLARI	208
E. PROFESSIONAL WRITE (PW) PROGRAMINA	209
GİRİŞ.....	209
F. ANA MENÜ	209
MAIN MENU	209

1. YARAT/DÜZELT	210
Create/Edit	210
2. DÜZENLE	210
Setup	210
1. BİRİNCİ YAZICI SEÇİMİ.....	211
Select Printer 1	211
2. İKİNCİ YAZICI SEÇİMİ	212
Select Printer 2	212
3. YAZICI KONTROL KODLARINI BELİRLEME	212
Specify Printer Control Codes.....	212
4. VERİ KAYIT ALANINI TANIMLAMA.....	212
Change Data Directory.....	212
5. ÇALIŞMA SÜRÜCÜSÜ TANIMLAMA	213
Change Work Drive	213
6. EKРАН RENKLERİNİ TANIMLAMA.....	213
Change Screen Colors	213
7. EKРАН GÖSTERİM HIZINI AYARLAMA.....	214
Change Screen Update Speed	214
8. INSERT TUŞUNU AÇMA/KAPAMA	214
Set default to insert/replace.....	214
3. ÇIKIŞ.....	214
Exit.....	214
G. WORKING COPY ALANI VE KLAVYE TUŞ.....	215
ÖZELLİKLERİ.....	215
YARDIM MENÜSÜ.....	215
F1 HELP Menu.....	215
H. DOSYA ve BASKI İŞLEMLERİ.....	220
F2 File/Print	220
1. DOSYA ÇAĞIRMA	220
Get file (Ctrl-G)	220
2. DOSYA KAYDETME.....	222
Save working copy (Ctrl-S).....	222
3. DOSYA SİLME.....	224
Delete file.....	224
4. DOSYA EKLEME.....	224
Insert file.....	224
5. YAZICIYA YAZDIRMA	225
Print working copy (Ctrl-O).....	225
6. BASKI ÖNİZLEME	227
Print preview (Ctrl-Prt Scr)	227
7. BELLEĞİ SİLME.....	231
Erase working copy.....	231

8. MAKRO KULLANIMI	231
Use macros (Alt-0).....	231
I. DÜZELTME.....	234
F3 Edit.....	234
1. SATIR EKLEME.....	235
Insert blank line (Ctrl-I).....	235
2. SÖZCÜK SİLME.....	235
Delete word (Ctrl-W).....	235
3. SATIR SİLME.....	235
Delete line (Ctrl-L).....	235
4. BLOK İŞARETLEME.....	235
Mark text (Ctrl-T).....	235
5. SÜTUN BLOKLAMA.....	238
Mark rectangle (Ctrl-R).....	238
6. BLOK YAPIŞTIRMA.....	239
Paste (Ctrl-P).....	239
7. SÖZCÜK KOYULAŞTIRMA.....	239
Boldface word (Ctrl-B).....	239
8. ALT ÇİZGİLİ YAZDIRMA.....	240
Underline word (Ctrl-U).....	240
9. ÇİZGİ ÇİZME.....	240
Draw lines (Ctrl-Y).....	240
10. BUL ve DEĞİŞTİR.....	241
A. Find & Replace (Ctrl-F).....	241
11. HESAP YAPMA.....	242
Calcute (Ctrl-M).....	242
J. BİÇİMLEME MENÜSÜ.....	243
F4-Format.....	243
1. SOL/SAĞ MARJ AYARI.....	244
Set left/right margins.....	244
2. ÜST/ALT BOŞLUK & SAYFA SATIR AYARI.....	245
Set top/bottom margins & length.....	245
3. DURAK YAPMA.....	245
Set tabs.....	245
4. SAYFA ÜZERİ, BAŞLIK YAZIMI.....	247
Set header.....	247
5. SAYFA ALTI, DİPNOT YAZIMI.....	248
Set footer.....	248
6. SATIR BAŞINI GEÇİCİ OLARAK DEĞİŞTİRME.....	248
Turn indent on/off (Ctrl-N).....	248
7. SATIR ARALIĞINI İKİ YA DA TEK YAPMA.....	248
Turn double spacing on/of (Ctrl-D).....	248

8. YAZIYI SATIRA ORTALAMA	249
Center line (Ctrl-X)	249
9. SATIRI SOLA YASLAMA	249
Left justify line	249
10 SATIRI SAĞA YASLAMA	249
A. Right justify line	249
K. YAZIM KLAVUZU	250
F5-Dictionary	250
1. Sözcük Kontrolü	250
Proof word (Ctrl-A)	250
2. Yanlış Sözcükleri Bulma	251
Proof document (Ctrl-V)	251
3. Eşanlımları Bulma	251
Find synonyms	251
L. ADRES MENÜSÜ	251
F6/Addresses	251
BÖLÜM VII	256
SAYI SİSTEMLERİ	256
A. İKİLİ SAYI SİSTEMİ (BINARY NUMBER SYSTEM)	256
1. İKİLİ SAYI SİSTEMİNDE DÖRT İŞLEM	257
2. İKİLİ SİSTEMDEN ONLUK SİSTEME GEÇİŞ	260
3. ONLUK SİSTEMDEN İKİLİ SİSTEME GEÇİŞ	261
B. ONALTILIK SAYI SİSTEMİ	262
1. ONALTILIK SAYI SİSTEMİNDE DÖRT İŞLEM	262
2. ONALTILIK SİSTEMDEN ONLUK SİSTEME GEÇİŞ	265
3. ONLUK SİSTEMDEN ONALTILIK SİSTEME GEÇİŞ	265
4. ONALTILIK SİSTEMDEN İKİLİ SİSTEME GEÇİŞ	266
5. İKİLİ SİSTEMDEN ONALTILIK SİSTEME GEÇİŞ	266
BÖLÜM VIII	268
PROGRAMLAMAYA GİRİŞ	268
A. OPERATÖRLER	268
1. Aritmetik İşlem Operatörleri	268
2. Mantık İşlemleri ve Operatörleri	269
3. İşlem Öncelik Sıraları ve Parantezli İşlemler	269
4. Karşılaştırma Sembolleri	269
B. PROGRAMLAMA İLE İLGİLİ GENEL BİLGİLER	270
1. PROGRAM, PROGRAMCI ve ANALİST	270
2. PROGRAMLAMA SAFHALARI	270
a. Problemi Tanıma	270
b. Çözüm Yollarını Tasarlama	271
c. Programı Kodlama	271
d. Programı Test Etme	271

e. Programı Belgeleme	271
3. ALGORİTMA	271
a. Algoritmanın Tanımı ve Kuralları	271
b. Algoritma Örnekleri	272
4. AKIŞ ŞEMALARI (DİYAGRAMLARI).....	275
a. Akış Şemasının Tanımı.....	275
b. Akış Şeması Sembolleri	275
c. Akış Şeması Örnekleri	276
5. BİLGİSAYARA DEĞERLERİN TANITIMI.....	281
a. Sabitler	281
b. Değişkenler	282
C. PROGRAMLAMA DİLLERİ	283
1. Alçak Düzeyli Diller	283
2. Yüksek Düzeyli Diller.....	283
BÖLÜM IX.....	286
BASIC PROGRAMLAMA DİLİ	286
A. BASIC DİLİNE GİRİŞ	286
1. Basic Dilinde Program Yazımı İçin Gerekli Ön Bilgiler:	286
2. ÖZEL KARAKTERLERİN AÇIKLAMALARI	287
B. BİLGİSAYARI AÇMA ve BASIC ORTAMINA	289
GEÇİŞ	289
C. BASIC'DE KLAVYE KULLANIMI	291
1. FONKSİYON TUŞLARI	292
2. EDİT TUŞLARI	296
BÖLÜM X	300
BASIC'İN TEMEL YAPISI	300
A. KOMUT TÜRLERİ	300
B. SATIR NUMARASI VERME	300
C. BİR ALT SATIRA GEÇME	301
D. SATIR EKLEME	301
E. SATIR SİLME.....	302
DELETE	302
F. PROGRAMI LİSTELEME	302
LIST	302
G. PROGRAMI DİSKE KAYIT ETME.....	303
SAVE	303
H. YENİ BİR PROGRAM YAPMAYA GEÇME	303
NEW	303
I. DİSKETTEN PROGRAM ÇAĞIRMA	304
LOAD	304
J. PROGRAMI ÇALIŞTIRMA.....	305
RUN	305

K. DOSYA LİSTESİNİ GÖRME	305
FILES	305
L. DOSYA ADI DEĞİŞTİRME	305
NAME ... AS	305
M. DİSKETTEN DOSYA SİLME	306
KILL	306
N. BASIC ORTAMINDAN DOS'A DÖNÜŞ (ÇIKIŞ)	306
SYSTEM	306
BÖLÜM XI	307
BASIC PROGRAMLAMA DİLİ	307
KOMUT ve DEYİMLERİ	307
A. EKTRAN SİLME, ÇALIŞTIRMA, AÇIKLAMA VE	307
SONA ERDİRME KOMUTLARI	307
1. EKTRANI SİLME	307
CLS	307
2. PROGRAMI ÇALIŞTIRMA	308
RUN	308
3. AÇIKLAMA YAZMA	308
REM	308
4. PROGRAMI SONA ERDİRME	308
END	308
5. PROGRAMI DURDURMA	309
STOP	309
B. ATAMA ve GİRİŞ/ÇIKIŞ KOMUTLARI	310
1. EKTRANA ve YAZICIYA YAZDIRMA	310
a. PRINT	310
b. LPRINT	315
c. WRITE	317
2. DEĞER YA DA İFADE ATAMA	317
LET	317
LET DEYİMİNİN ÖZELLİKLERİ	319
3. KLAVYEDEN BİLGİ GİRME	320
INPUT	320
4. PROGRAM İÇERİSİNDEN VERİLERİ İŞLEME	324
READ-DATA	324
READ-DATA DEYİMLERİNİN ÖZELLİKLERİ	325
RESTORE	327
C. EKTRAN DÜZENLEME KOMUTLARI	328
1 SÜTUN BELİRTEREK YAZDIRMA	328
TAB	328
2. SATIR ve SÜTUN BELİRTEREK YAZDIRMA	329
LOCATE	329

3. SAYISAL DEĞERLERİ İSTENEN KALIPTA YAZDIRMA	330
PRINT USING	330
D. SİSTEM KOMUTLARI	333
1. SATIRLARDA DÜZELTME YAPMA	333
EDIT	333
2. SATIR NUMARALARINI YENİDEN DÜZENLEME	334
RENUM	334
3. OTOMATİK SATIR NUMARASI VERME	335
AUTO	335
4. SESLİ UYARMA	335
BEEP	335
5. MS-DOS'A GEÇİCİ OLARAK ÇIKMA	335
SHELL	335
6. MS-DOS'DAN BASIC'E GERİ DÖNÜŞ	336
EXIT	336
E. KONTROL KOMUTLARI	336
1. KOŞULSUZ GÖNDERME	336
GOTO	336
2. KOŞULLU SAPMA	339
IF-THEN-ELSE	339
3. BASIC DİLİNDE BAĞLAÇLAR	341
AND-OR-NOT	341
(VE-VEYA-DEĞİL)	341
4. ALT PROGRAMLAR (SUBROUTINE)	344
a. GOSUB-RETURN	344
b. ON...GOTO / ON...GOSUB DEYİMLERİ	346
F. SAYAÇ KULLANIMI	349
G. DÖNGÜ KOMUTLARI	352
1. ÇEVİRİM VE DÖNGÜLER	352
a. FOR...NEXT DÖNGÜSÜ	352
1) FOR/NEXT İçinde READ/DATA Uygulamaları	355
2) İççe Döngüler	356
b. WHILE...WEND DÖNGÜSÜ	359
c. WHILE NOT...WEND	359
H. DİZİNLİ DEĞİŞKENLER ve BOYUT KULLANIMI	360
1. DIM	360
2. OPTION BASE	360
a. TEK BOYUTLU DİZİLER	361
1) Sayısal Dizi Açma	361
2) Alfasayısal Dizi Açma	361
b. ÇİFT BOYUTLU DİZİLER	362
1) Sayısal Dizi Açma	363

2) Alfayısal Dizi Ama	363
3. BELLEK ve BOYUT SIFIRLAMA.....	364
a. CLEAR.....	364
b. ERASE.....	364
I. HAZIR FONKSİYONLAR	365
1. MATEMATİKSEL FONKSİYONLAR	365
a. INT	365
b. FIX	365
c. CINT	365
d. LOG	366
e. SIN	366
f. COS	366
g. SQR.....	366
h. TAN	367
i. ATN	367
j. ABS.....	367
k. CDBL.....	367
l. CSNG.....	368
m. EXP.....	368
n. SGN	368
o. RND.....	368
2. KARAKTER FONKSİYONLAR	369
a. CHR\$	369
b. STRING\$	369
c. LEN.....	370
d. LEFT\$.....	370
e. MID\$	370
f. RIGHT\$.....	371
g. SPACE\$	371
h. VAL	371
i. STR\$	372
j. SWAP	372
k. INSTR.....	373
l. ASC.....	374
m. HEX\$	374
n. OCT\$	374
3. İMLEÇ HAREKET FONKSİYONLARI.....	375
CSRLIN DEĞİŐKENİ ve POS FONKSİYONU	375
4. KARAKTER VERİ GİRİŐ FONKSİYONLARI	375
a. INKEY\$	375
b. INPUT\$.....	375
c. LINE INPUT	376

5. TARİH ve SAAT FONKSİYONLARI	376
a. DATES	376
b. TIMES	376
6. GRAFİK FONKSİYONLARI	377
a. SCREEN	377
b. WIDTH	377
c. WIEW	377
d. WIEW PRINT	378
e. WINDOW	378
f. LINE	378
g. CIRCLE	379
h. DRAW	380
i. PSET	381
j. PRESET	382
k. COLOR	382
l. POINT	383
m. PAINT	383
n. GET (Grafik)	383
o. PUT (Grafik)	383
7. SES FONKSİYONLARI	384
a. PLAY	384
b. SOUND	385
8. EKРАН DÜZENLEME FONKSİYONLARI	385
a. WIDTH	385
b. COLOR	386
c. LOCATE	387
8. DOSYA BİRLEŞTİRME DEYİMLERİ	388
a. MERGE	388
b. CHAIN	388
9. DİĞER KOMUT, DEYİM ve FONKSİYONLAR	389
a. RESET	389
b. MKDIR	390
c. CHDIR	390
d. RMDIR	390
e. DEF FN	391
f. DEF	391
g. ERR ve ERL	392
h. ON ERROR GOTO-RESUME	392
i. FRE	393
j. PEEK	393
k. POKE	393
l. KEY	394

m. ON KEY	395
J. DOSYA DÜZENLEMESİ	396
1. SIRALI ERİŞİMLİ DOSYALAR	396
a. SIRALI DOSYALARDA KULLANILAN KOMUTLAR	396
1) DOSYA AÇMA DEYİMİ	397
OPEN	397
2) DOSYA KAPAMA DEYİMİ	398
CLOSE #	398
3) BİLGİ KAYIT DEYİMİ	398
PRINT #	398
4) BİLGİ KAYIT DEYİMİ	399
WRITE #	399
5) BİLGİ OKUMA DEYİMİ	399
INPUT #	399
6) DOSYA SONU KONTROL FONKSİYONU	399
EOF	399
7) KAYIT UZUNLUĞU BELİRTME FONKSİYONU	400
LOF	400
8) SON KAYIT BELİRTME FONKSİYONU	400
LOC	400
9) KARAKTER DİZİSİNİ KAYIT DEYİMİ	400
LINE INPUT #	400
b. SIRALI ERİŞİMLİ DOSYA UYGULAMALARI	401
2. DOĞRUDAN (RANDOM) ERİŞİMLİ DOSYALAR	407
a. DOĞRUDAN ERİŞİMLİ DOSYALARDA KULLANILAN	407
KOMUTLAR	407
1) DOSYA AÇMA DEYİMİ	408
OPEN	408
2) BUFFER'İ DEĞİŞKENLERE PAYLAŞTIRMA	408
FIELD	408
3) KAYIT DEYİMİ	409
PUT #	409
4) BİLGİ OKUMA DEYİMİ	409
GET #	409
5) SAYILARI DÜZENLEME FONKSİYONLARI	410
MKİ\$, MKS\$, MKD\$	410
6) BİLGİLERİ SOLA-SAĞA DAYALI KAYIT	410
LSET-RSET	410
7) KARAKTER BİLGİYİ SAYISAL BİLGİYE DÖNÜŞTÜRME	410
CVI, CVS, CVD	410
8) DOSYA KAPATMA DEYİMİ	411
CLOSE	411

b. DOĞRUDAN ERİŞİMLİ DOSYA UYGULAMALARI	411
K. BASIC PROGRAMLARININ DERLENMESİ	421
ÖRNEK PROGRAMLAR	423
BÖLÜM XII	440
EK BİLGİLER	440
A. BASIC PROGRAMLAMA DİLİNDE EKRANA	440
ÇIKABİLECEK HATA MESAJLARI	440
B. MS-DOS HATA MESAJLARI	442
C. ASCII, HEX ve KARAKTER KODLARI LİSTESİ	447
BİLGİİŞLEM TERİMLERİ SÖZLÜĞÜ	449
İNDEKS	453
KAYNAKÇA	462

BÖLÜM I.

BİLGİSAYARA GİRİŞ

A. BİLGİSAYARIN TANIMI

Bilgisayar: Verileri sayısal olarak işleyip, belli bir mantık doğrultusunda programlanabilen, aynı mantıkla çok sayıda veriyi depolayıp, işleyebilen ve sonuca ulaşabilen elektronik makinelerdir.

Bu tanımdan sonra, şunu unutmamalıyız: Bilgisayarın tanımında ne kadar bu yeteneklerini saydıysak da insan katkısı olmadan, hiçbir şeyi yapamaz. Bilgisayarı yapan ve programlayan insandır. Tanımda da belirttiğimiz gibi, bizim bilgisayara herhangi bir bilgi vermeden, ondan bilgi istememiz ve herşeyi bilmesini beklememiz yanlış olur. Bilgisayarlar bizim verdiğimiz bilgiler doğrultusunda çalışır ve görev yaparlar.

Bilgisayarın tanımından sonra bilgi, bilgi işlem ve veri nedir? Bunları öğrenelim:

Veri: İletişim, açıklama ve işlem yapabilmek için kullanılan sayı, harf ve simgelerdir.

Bilgi: Bilgisayarda bilginin tanımı şu biçimde yapılabilir: Üzerinde işlem yapılan Veri ögesi ya da değerlerin, bilgi işlem yardımı ile yararlı ve anlamlı bir biçime sokulmuş biçimidir.

Bilgişlem: Verinin, bilgi şekline getirilme süreci ve bilgi üzerinde çeşitli işlemlerin yapılmasıdır.

Bilgi ve Veri Arasındaki Farklar:

- Veri bilgi işleme konu olan kısım, bilgi ise işlemin sonucudur.
- Veri işlenmemiş bilgidir. Bilgi, verilerin bilgi işlem yardımıyla yararlı ve anlamlı duruma getirilmiş sonucudur.

B. BİLGİSAYARIN TARİHSEL GELİŞİMİ

İnsanoğlu, işlerini daha kolay, daha pratik ve daha az emek harcayarak yapmanın yollarını sürekli aramış; kendi görevini, çeşitli araç ve hayvanlara yüklemeye çalışmıştır. Bu çalışma ve merak, gittikçe artan bir istek ve hızla sürmüştür. Her buluş bir buluşu çağırıştırılmış, yeni oluşumlara yol açmıştır. İnsanın bitmek bilmeyen isteği ve azmi ile teknoloji de sınırsız bir hız kazanmıştır. Bu konuda en büyük buluş ve gelişmelerden biri de bilgisayardır.

Dünya kurulduğundan bu yana süregelen bu araştırmalar ve çalışmalar sonucu, sayısal hesap, bilgi birikimi ve deposu, kayıt merakı bugünkü bilgisayarların zeminini hazırlamıştır.

Bilgisayarın tarihsel gelişimine bakacak olursak, kesin bir sınıflandırma yapamamakla birlikte tarihsel gelişimini beş kuşak altında inceleyebilir ve sınıflandırabiliriz.

Bu konuda geniş bir açıklamaya girmeden önce çok basit olarak, anlayabilmek için sayma boncuğunu da (Abaküs), basit bir bilgisayar olarak görürsek bilgisayarların nasıl bir değişime uğradığını düşünebiliriz.

1. BİRİNCİ KUŞAK BİLGİSAYARLAR. (1945-1956)

İlk kuşak bilgisayarlar, çok büyük hacimli ve çok enerji ve ısı harcayan bilgisayarlardı. Bunlar **vakum tüpler** ve **radyo lambaları** ile çalışmaktaydı.

Bu konuda ilk bilgisayar adını verebileceğimiz âlet **1945** yılında **ENIAC** (**E**lectronic **N**umerical **I**ntegrator **A**nd **C**alculator)'dır. **40 ton** ağırlığında büyük hacimli ve **18000** vakum tüple çalışmaktaydı. Bu bilgisayar **Pennsylvania Üniversitesi**'nde yapılmış ve başarılı bir şekilde çalıştırılmıştır.

1946 yılında **ENIAC**'ın başarısından sonra **EDVAK** adında yeni bir bilgisayar yapıldı.

İlk ticarî bilgisayar, **1951** yılında, **UNIVAC-1** adıyla üretildi.

Bu süreler içinde yapılan bilgisayarlar teknik bakımdan birbirinin aynıydı diyebiliriz. Şimdiki bilgisayarlara göre hız ve işlem kapasitesi açısından çok yavaş ve düşüktüler. Bu kuşak bilgisayarlar **10000-20000** kadar karakter depolayabilmekteydi.

2. İKİNCİ KUŞAK BİLGİSAYARLAR. (1957-1964)

Bu yıllardan sonra, **Nobel Ödülü**'ne değer görülen **Transistör**'ün **1956** yılında kullanıma başlamasıyla bilgisayarlar, hacim açısından küçülürken, hız açısından artış sağlamaktaydı. Bu türden ilk bilgisayar, yani transistörlü ilk bilgisayar **1957** yılında **Burroughs Şirketi** tarafından **Hava Kuvvetleri** için yapıldı.

Birinci kuşak bilgisayarların sonunu getiren, teknolojik olarak daha ileri düzeydeki bu bilgisayarlar, yukarıda saydığımız özelliklerinin yanısıra hızlı giriş-çıkış, yazılım ve programlama tekniklerinin karmaşıklığı açısından ayırıcı özelliklerdir.

3. ÜÇÜNCÜ KUŞAK BİLGİSAYARLAR. (1965-1970)

Bu kuşak bilgisayarlarda transistörlerin yerini **Entegre Devre**'ler almıştır. Daha etkin giriş-çıkışa sahip bu bilgisayarlar, disk, bellek gibi donanımlara da sahipti.

Entegre devre sistemiyle, bilgisayarların hacimleri daha da küçülürken, hız ve bellek açısından çok iyi sonuçlar alınmaktaydı. Çoklu işlem, çoklu programlama, görüntü bellek (**Virtual Memory**) bu kuşak bilgisayarların ürünüdür.

4. DÖRDÜNCÜ ve BEŞİNCİ KUŞAK BİLGİSAYARLAR. (1970 ve Sonrası)

Bilgisayarın tarihî gelişimine dikkat edersek, son **30-40** yılda büyük gelişmeler olmuş ve hacim ve maliyet küçülürken, bellek ve hız açısından artış olmuştur.

Günümüzde bilgisayarlarla her alanda karşılaşmaktayız. Dördüncü kuşak bilgisayarlar daha çok alana girmeleriyle ve işlevlerinin artmasıyla tanınmaktadır. Günlük yaşamımızda karşılaştığımız birçok alanda, işlerimizi daha kısa sürede ve daha güvenilir biçimde bilgisayarlara bırakır olduk.

Günümüzde **5. Kuşak Bilgisayar** bulunmamakla beraber, geliştirme çabaları sürmektedir. Bu kuşakta yapılmak istenen işlem hızını artırmak, kullanımı kolaylaştırmak, daha karmaşık yazılımları destekleyebilmesini sağlayabilmektir. Ulaşılmak istenen en önemli amaç ise **Yapay Zeka** (**AI-Artificial Intelligence**) olmaktadır.

C. BİLGİSAYARLARIN KULLANIM ALANLARI

Bilgisayarları kullanım alanları açısından saymak güçtür. Günümüzde bilgisayarların hizmete girmediği alan kalmamıştır. Günlük yaşamımızda her yerde, her an karşımıza çıkmaktadırlar. Bu açıdan evde, okulda, büroda, hastanede, fabrikada, bankada ve akla gelebilecek birçok yerde kullanılmaktadır. Örnek olarak aşağıda, kullanıldığı alanlardan birkaçını sayalım:

- * Endüstride; kalite kontrol ve montajda,
- * Eğitimde; değişik anlatım yöntemleri ve araştırmalarda. Deneysel anlatımları gözlemlene ve uygulamada,
- * İşletmelerde; muhasebe, stok kontrol, müşteri takibi, evrak takibi, bordro işlemlerinde,
- * Mühendislikte; tasarımda, statik hesaplamada ve üç boyutlu çizimde,

- * Masa üstü yayıncılıkta; gazetelerin dizgi, baskı, sayfa düzeni gibi alanlarında,
- * Elektrik, su ve PTT faturalarının düzenlenmesinde,
- * Bankalarda; havale, mevduat gibi hesap akışlarında,
- * Bilimsel arařtırmalarda; uzay, tıp, fizik, biyoloji gibi alanlarda,
- * Sosyal ve ekonomik arařtırmalarda; istatistik hazırlamada,
- * Evlerde kullanılan; televizyon, çamařır makinesi, elektrik fırını gibi aletlerin programlanmasında,
- * Çeviri, ansiklopedik bilgi, kitap yazımında,
- * Çeřitli kontrol mekanizmalarında kullanılır.

Yukarıda sayamadığımız daha birçok yerde, bilgisayar kullanımı yaygındır.

D. BİLGİSAYAR TÜRLERİ

Bilgisayarlar önce ikiye ayrılırlar:

1. Çalışmalarına göre bilgisayarlar.
2. Büyüklüklerine göre bilgisayarlar.

1. Çalışmalarına Göre Bilgisayarlar:

a. Sayma Tekniğine Dayalı Bilgisayarlar:

Bunlara **Digital Bilgisayarlar** da denir. Sayısal veri kullanarak, sayısal olarak işleyip, sayısal olarak sonuca ulaşırlar.

b. Ölçme Tekniğine Dayalı Bilgisayarlar:

Analog Bilgisayarlar da denir. Fiziksel büyüklükleri veri olarak alan ve sonucu başka bir fiziksel güç olarak veren, verilerle fiziksel büyüklükler arasında yapılan örneklemelere dayanan bilgisayarlardır. Veri bir grafik olarak verilmişse grafik olarak sonuç alınabilir. Kısaca girişte, veri nasıl verilmişse, o şekilde çıktı olarak alınabilir. Daha çok kontrol işlemlerinde kullanılırlar. Basınç, sıcaklık, voltaj gibi fiziksel ve elektriksel değerleri ölçmede kullanılırlar.

c. Karma Tip Bilgisayarlar:

Hybrid ya da **Melez Bilgisayarlar** da denilir. Sayısal ve örneksel bilgisayarların karışımından oluşan bilgisayarlardır. Bu tip bilgisayarlarda örneksel ve sayısal üniteler arasında bilgi alışverişi dönüştürücüler aracılığı ile sağlanır.

2. Büyüklüklerine Göre Bilgisayarlar

Bilgisayarlar ana bellek, disk/disket, terminal, yazıcı bağlantıları ve hız açısından sınıflandırılırlar. Değişik kapasitedeki bilgisayarları açıklamak amacı ile bir sınıflandırma yapmak gerekirse, kesin olmamakla birlikte 4'e ayrılır. Çünkü bazı eklentilerle bir mikro bilgisayar, ana bilgisayara dönüştürülebilir. Bir de bu bilgisayarların birden fazla kombinasyonunu kullanabilen bilgisayarlar vardır. Bunları sınıflandırmak güçtür. Süpermikro, süpermini gibi...

Şu şekilde bir sınıflandırma yapabiliriz:

1. Mikro Bilgisayarlar.
2. Mini Bilgisayarlar.
3. Ana Bilgisayarlar.
4. Süper Bilgisayarlar.

E. BİLGİSAYAR KAVRAMLARI

Bilgisayarların çalışma yapısı ve donanımında kullanılan başlıca kavramları inceleyerek tanıyalım:

1. BIT, BYTE, K.BYTE, M.BYTE, G.BYTE, T.BYTE

Bilgisayarlar ikili sayı sistemine göre çalışırlar. Karakterler **8** tane, **0** ve **1**'den oluşan, sayılardan oluşmuştur. Bu **0** ve **1**'lerin her birine **Bit** adı verilir. **Bit** kısaca en küçük bilgi birimidir. **B**inary **D**igit (ikili sayı) sözcüklerinin ilk harflerinden oluşmuştur.

1 byte **00000000** ile **11111111** arasında değişir. Karakterler şu şekilde olurlar: **01011011, 11011011, 10111011** gibi... Yanyana **8 bit**'in **256** tane kombinasyonu vardır.

8 bit uzunluğundaki, ikili bit kombinasyonlarına **byte** denir. Kısaca **8 bit, 1 byte** eder diyebiliriz. Her byte **8 bit**'ten , her karakter **1 byte**'tan oluşur.

Bu da **1 karakter = 8 bit** ya da **1 byte**'dir.

Byte'ın bir üst birimi **Kilo Byte**'tır (**Kbyte**). **1024 byte=1 kbyte** eder. Bu da 2'nin bine en yakın kuvvetidir.

$$2^{10} = 1024 \text{ eder.}$$

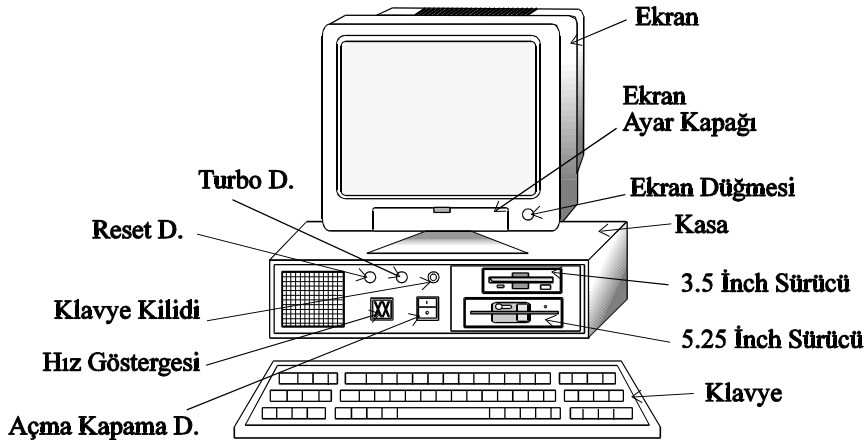
Kbyte'ın bir üst birimi de, **Mega Byte**'tır. (**Mbyte**) **1024 Kbyte=1 Mbyte** eder.

Toplu olarak bir kez daha görelim:

8 BIT	=	1 BYTE
1024 BYTE	=	1 KBYTE
1024 KBYTE	=	1 MBYTE
1024 MBYTE	=	1 GBYTE (Giga Byte)
1024 GBYTE	=	1 TBYTE (Tera Byte) eder.

2. DONANIM (HARDWARE)

Bilgisayar sistemini oluşturan mekanik ve elektronik (fiziksel) araçlara **donanım** adı verilir. Bunlar Merkezi İşlem Birimi (M.İ.B), klavye, yazıcı, ekran vb. fiziksel ünitelerdir.



Bilgisayar ve Donanımları

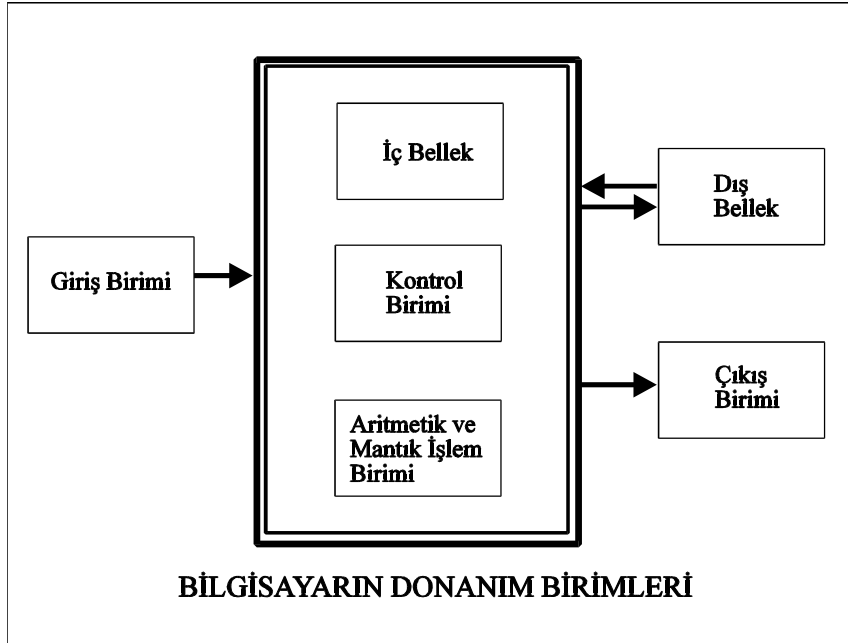
Donanım birimleri ikiye ayrılır:

a. Dış Donanım Birimleri

- 1) Giriş Birimi
- 2) Çıkış Birimi
- 3) Dış Bellek Birimi

b. İç Donanım Birimleri

- 1) Aritmetik ve Mantık Birimi
- 2) Kontrol Birimi
- 3) Bellek Birimi



3. YAZILIM (SOFTWARE)

Bilgisayarın çalışabilmesi için hazır olan yazılı programlardır.

Üçe ayrılır:

- a. Sistem Yazılım Programları
- b. Yorumlayıcı (Interpreter) ve Derleyici (Compilers) Yazılım Programları.
- c. Kullanıcı (Uygulama) Yazılım Programları

a. Sistem Yazılım Programları: Bilgisayarın açılması, diğer yazılım programlarının devreye sokulması ve bağlı birimlerin çalışmalarını yönlendiren girdi ve çıktıları belirli bir şekilde düzenleyen, verilen komutları algılamasını ve kullanıcı ile bilgisayar arasında bağlantıyı kuran yazılı programlardır. Kısaca yönetici ve denetleyici olarak tanımlayabiliriz. Örnek olarak **CP-M, MS-DOS, UNIX**'i verebiliriz.

b. Yorumlayıcı (Interpreters) ve Derleyici (Compilers) Yazılım Programları: Kaynak programları veri olarak okuyan ve yorumlayıcının dil kurallarına göre doğru yazılıp yazılmadığını kontrol eden, bunları uygun ise makine diline dönüştüren ve işleten yazılımlardır. **BASIC Yorumlayıcısı, COBOL, PASCAL Derleyicisi** gibi.

c. Kullanıcı (Uygulama) Yazılım Programları: Verileri işleyerek istenen sonuçları veren programlardır. Bu programlar bir yorumlayıcı veya derleyici programların dillerine göre yazılır. Bu tür programlar herkesin kolaylıkla kullanabileceği basit programlardır. **Muhasebe, Stok Kontrol, Adres Takip, Çek-Senet Takip Programı** gibi. Bu yazılımlar RPG, PL/1, BASIC, PASCAL, FORTRAN, COBOL gibi dillerle yazılırlar.

4. İŞLETİM SİSTEMİ

İşletim Sistemi: Bilgisayarın açılışından kapanışına kadar aktif durumda olan, kullanıcı ile bilgisayar arasındaki bağlantıyı sağlayan, komutları algılamaya, programları ve çevre ünitelerini kullanılır duruma getiren, sistem yazılım programlarıdır.

İşletim sistemi, giriş birimlerinden aldığı komutlara göre bellek ile giriş-çıkış birimleri arasında alışverişi, programların çalıştırılmasını ve dosya işlemlerinin gerçekleştirilmesini sağlayan ve denetleyen programdır. Kısaca işletim sistemi; bilgisayarın can damarıdır.

Bilgisayarların tiplerine, modellerine göre kullandıkları işletim sistemi farklıdır.

SORULAR

1. Bilgisayarın tanımını yapınız.
2. Veri ve Bilgi nedir? Aralarındaki farkları sayınız.
3. Bilgisayarlar kaçaya ayrılır? Sayınız.
4. Donanım nedir? Kaçaya ayrılır? Açıklayınız.
5. Donanım Birimleri Şemasını çizin ve anlatınız.
6. Yazılım kaçaya ayrılır? Kısaca açıklayınız.
7. Bilgisayarlar kaç çeşittir? Kısaca açıklayınız.
8. BIT, BYTE, KBYTE ve MBYTE kavramlarını açıklayınız.
9. 64 KBYTE kaç BYTE eder?
10. 32768 KBYTE kaç MBYTE eder?
11. İşletim Sistemi nedir? Tanımlayınız.

BÖLÜM II.

BİLGİSAYARIN GENEL YAPISI

A. GİRİŞ BİRİMİ

Bilgisayarın işlemesi için veri ve bilgilerin girildiği birimdir. Giriş ünitesi aracılığıyla girilen bilgiler bilgisayar tarafından algılandıktan sonra bellekte tutulur. Giriş üniteleri olarak en çok kullanılan araçlar **Klavye, Disk/Disket, Teyp, Scanner**'ı sayabiliriz.

B. MERKEZİ İŞLEM BİRİMİ (M.İ.B.)

(**C**entral **P**rocessing **U**nit = C.P.U) Bilgisayarın en önemli bölümüdür. Bütün bilgiler bu bölüm aracılığı ile kontrol edilir ve denetlenirler. Matematiksel ve mantıksal işlemler bu bölümden onay aldıktan sonra işleme girerler. Bilgiler bu birim aracılığıyla bellekte tutulur, diğer birimlere gönderilir.

Merkezi İşlem Birimi üçe ayrılır:

1. Aritmetik ve Mantık Birimi
2. Kontrol (Denetim) Birimi
3. Bellek Birimi

1. ARİTMETİK ve MANTIK BİRİMİ

Dört işlem ve mantıksal işlemlerin yapıldığı birimdir. Matematiksel ve mantıksal işlemler bu birim tarafından analiz edildikten ve onaylandıktan sonra belleğe aktarılır.

2. KONTROL (DENETİM) BİRİMİ

Aritmetik ve Mantık Birimi tarafından gönderilen bilgilerin kontrol edildiği birimdir. Bu birim aynı zamanda diğer birimlerle alışverişi de düzenler. Bilgi alışverişini sağlar.

3. BELLEK BİRİMİ

Verilen bilgi ve programların saklandığı ve tutulduğu yerdir. Ana bellek, disk, disket, teyp gibi.

Bilgisayar tarafından verilerin işlenebilmesi ve saklanabilmesi için bellekte birtakım bilgilerin bulunması gerekir. Bu bakımdan iki tür bellek vardır:

- a. RAM Bellek
- b. ROM Bellek

a. RAM BELLEK (RANDOM ACCESS MEMORY): (Rastgele Erişimli Bellek). Ana belleğin diğer adıdır. İşletim sistemi ve her türlü programın girdiği ve kullanıcı tarafından silinip değiştirilebilen bellektir. Bilgisayardan akım kesildiği an içindeki bilgiler silinir.

MS-DOS normalde ana belleğin **640K**'lık kısmını okuyabilir. Bu da çoğu uygulama programlarında yetersiz kalmaktadır. Bunun için belleğin üst kısımları da çeşitli uygulamalarla kullanılabilir duruma getirilir. Belleğin katmanlarını ve adlarını inceleyelim:

Kullanılabilir Bellek (Conventional Memory): 640K boyutundadır. DOS'un kullandığı ve TSR (bellekte kalıcı) programların bulunduğu bölümdür.

Eklenebilir Bellek (Extended Memory (XMS)): Kullanılabilir belleğin üzerine eklenir. 80286 ve üstü bilgisayarlarda uygulanabilir. Belleği **384K** arttırarak **1Mb** kullanılabilir duruma getirir.

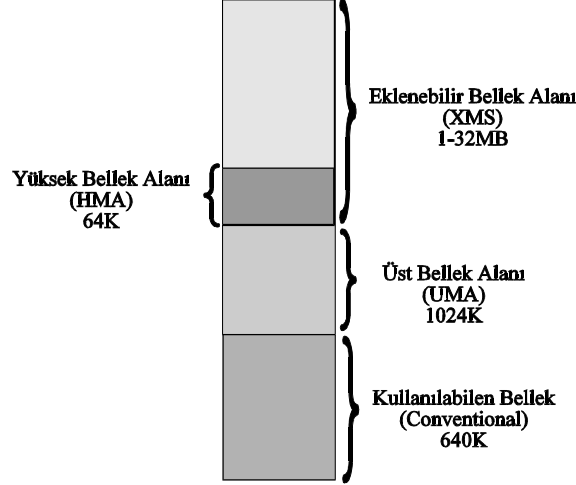
Genişleyebilir Bellek (Expanded Memory (EMS)): **640K** üzerine bellek eklemenin bir başka çeşididir.

Üst Bellek Alanı (Upper Memory Area (UMA)): 80386 ve üzeri bilgisayarlarda kullanılabilir. Bu bellek eklendiğinde DOS bazı programları bu bölüme atarak çalıştırır.

Yüksek Bellek Alanı (High Memory Area (HMA)): Eklenebilir belleğin ilk **64K**'lık alanıdır.

Yukarıda sözünü ettiğimiz bellek çeşitlerini kullanabilmek için, DOS'un bu konudaki programlarını ayarlamamız gerekir.

Belleğin bölümlerini aşağıdaki çizelgede görelim:



Belleğin Bölümleri

b. ROM BELLEK (READ ONLY MEMORY): (Yalnız Okunur Bellek) İç bellektir. Üretici firma tarafından bilgiler bellek çipinin içine yerleştirilir. Bilgisayarı ilk açtığımızda ekranda beliren görüntüler ve işletim sisteminin bilgisayara yüklenmesi için gerekli programlar **ROM**'un içine yerleştirilir. Daha önce depo edilmiş programı ana belleğe okur. Kullanıcı tarafından içindeki bilgiler değiştirilemez. Bilgisayarın kalıcı belleğidir. Akım kesilmesi sonrasında bellekteki bilgiler kalıcıdır. Bilgisayarı yeniden açtığımızda yüklenir. İçindeki bilgiler işletim sistemi ve özel dillerle (**Assembler, C** gibi...) okunabilir.

ROM Bellek kendi arasında ikiye ayrılır:

Birincisi **PROM (Programmable ROM - Programlanabilir ROM)**: Programcı tarafından içine bilgi yüklenebilir ve bir daha değiştirilemez. Bir kez yazılıp bir daha değiştirilmeyecek yazılımlar için kullanılır.

İkincisi de **EPROM (Erasable PROM - Silinebilir PROM)**: Daha çok revizyonlar gerektirecek ve çok sık değişmeyecek yazılımlar için kullanılır.

C. ÇIKIŞ BİRİMİ

Bilgisayarın verilen bilgileri işleyip, elde ettiği sonucu verdiği birimdir. Ekran, yazıcı, disk/disket, kart, şerit delici vb.

Çıkış birimleri hakkında daha geniş bilgiyi ilerideki konularda göreceğiz.

D. YARDIMCI BELLEK BİRİMİ

Bilgisayarda yazdığımız bilgiler önce ana bellek tarafından tutulur. Ana belleğin alacağı bilgilerin sınırlı olması ve bilgisayarı kapattığımızda, bilgilerin silinmesi daha sonra bu bilgileri kullanmamızı olanaksızlaştırıyordu. İşte bu nedenle bilgisayara girdiğimiz bilgileri yardımcı bellek birimlerine kayıt ederek gerektiğinde kullanabiliriz.

Yardımcı bellek denince akla en çok kullanılan **disk** ve **disket** gelir. Bunların dışında **teyp**, **CD ROM**'u (Compact Disk - Read Only Memory) sayabiliriz. Bu tür bellek birimleri daha çok bilgi aldıkları ve istenildiği zaman yeniden okuyup güncelleştirilebildiği için kullanılmaları kaçınılmazdır.

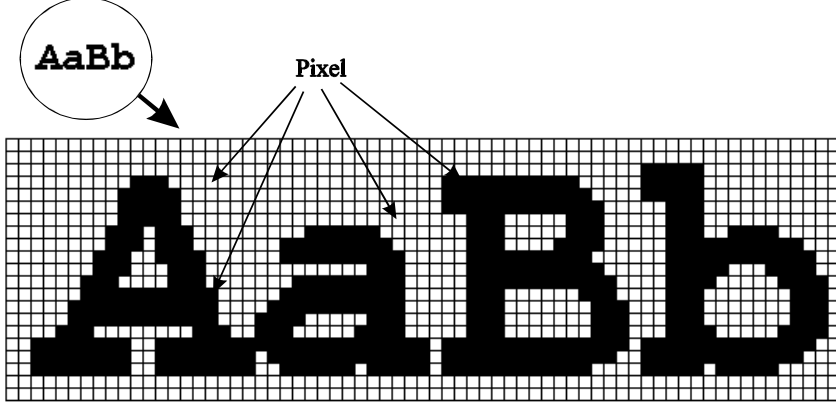
E. GİRİŞ VE ÇIKIŞ BİRİMLERİ

Daha önce giriş ve çıkış birimleri hakkında kısaca bilgi vermiştik. Bu birimler hakkında daha geniş bilgileri sırasıyla göreceğiz.

1. EKРАН VE EKРАН ÇEŞİTLERİ

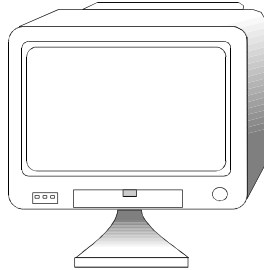
Bilgisayarın çıkış birimidir. Monitör olarak da bilinir. Bilgisayara verdiğimiz bilgileri, isteğe göre ekrandan yansıtır. Boyutu çoğunlukla **12"** ya da **14"**'tir. **24 satır**, **40-80** sütundan oluşur. **25. satır** kullanımı isteğe bağlıdır. Ekranda görebileceğimiz en küçük noktaya **Pixel** adı verilir. Pixeller çoğaldıkça daha net görüntü elde edilir. Bilgisayarda gördüğümüz harf, sembol, çizim vb. herşey noktalardan oluşur.

Renkli (**Color**) ya da Renksiz (**Monochrome**) olabilirler. Çeşitli grafik özellikleri ile grafik çizebilme özelliğine sahiptirler. Grafik özelliği bilgisayarın grafik kartını da ilgilendirir. Grafik kartları **Hercules**, **CGA**, **EGA**, **VGA**, **SVGA** vb. kartlardır. Grafik kartları, monitör ile uyumlu olmalıdır.



Harflerin Ekranında Görünümü

Monochrom ya da Color renk anlamındadır. **W/B (White/Black)** Siyah/Beyaz ekranı belirtir. Ayrıca **50 hz**, **60 hz** gibi, frekansları da belirtilir. Ekranların sınıflandırılması üç özelliğe göre yapılır: Ekranın saniyede kaç kez görüntülediğine yenilenme oranı ya da yatay tarama frekansı denir. Bu **hertz (Hz)** ile ölçülür. Örneğin saniyede **70 Hz**, **70 ekran** yenilenme oranı demektir. Bir monitörün, temiz görüntülü olması ve **CRT** üzerinde saniyede en az **60** tam ekran görüntü boyayarak titremeden çalışması gerekmektedir. Bu durum gözlerimizi de korur. Devre kartı alırken de aynı özellikte (Hz) olmasına dikkat edilmelidir.



Ekran

1. Boyut: 5 ile 25 inch arasında değişir. En çok kullanılanları, 12" ve 14" olanlardır. 25 satır, 80 sütundan oluşur.

2. Renk: Tek renkli olan monitörler siyah-beyaz, yeşil ve amber'dir. 2 renkten 16 milyon renk tona varan monitörler bulunmaktadır.

3. Çözünürlük: Monitörün görüntüsü ile kalitesi ölçülür. Ne kadar net görüntü sağlanabiliyorsa o kadar iyidir. Netliğin iyi olması da pixel sayısına yani adreslenebilir nokta sayısına bağlıdır. Bu sayı 65.000'den 16.000.000'a kadar değişebilir. **Maximum Yatay Nokta Sayısı x Maximum Dikey Nokta Sayısı** formülü ile hesaplanır.

Daha önce de belirttiğimiz gibi pixel ekrandaki en küçük nokta idi. Bu noktalar ne kadar çok olursa, görüntü de o kadar net olur. (720 X 350), (640 X 350), (800 x 600), (1024 x 728) gibi...

CGA (Color Graphics Adapter): 320x200 640x200 çözünürlüktedir. 4 renk gösterir.

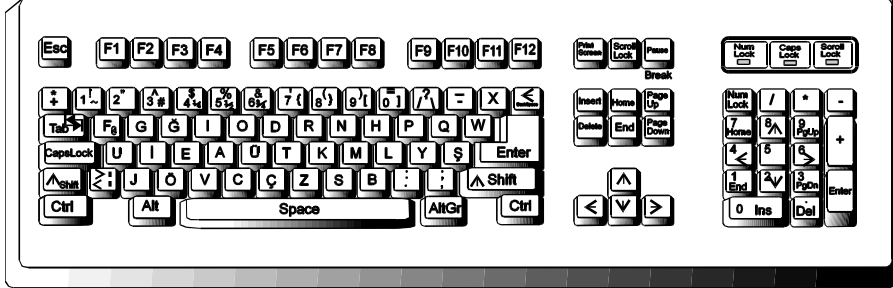
EGA (Enhanced Graphics Adapter): 640x350 720x350 çözünürlüktedir. 16 renk gösterir.

VGA (Video Graphics Array): 720x400 çözünürlüktedir. 16 renk ve 320x200 çözünürlükte 256 renk gösterebilir. **Süper VGA** kartlarla 1024x768 ve 1280x1024 çözünürlükte görüntü elde edilir. **VGA** kartının kendi belleği vardır. Bellek arttıkça renk sayısı artar.

2. KLAVYE ve KLAVYE ÇEŞİTLERİ

Klavye: Program yazımı, veri ve komut girişinde en çok kullanılan giriş birimidir. Sayısal, alfabetik ve özel karakterlerin bulunduğu tuşlar topluluğudur. Girilen bilgiler tuşa basılınca o tuşun ilgili devresini harekete geçirerek, tampon belleğe (buffer) alınır. Sonra kullanıcının görmesi için ekrana yansıtılır.

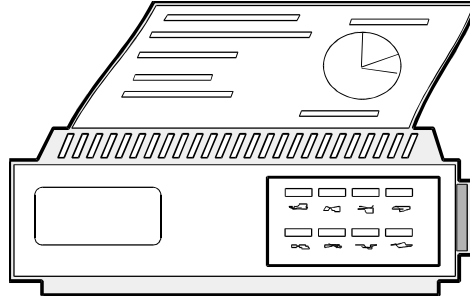
Klavyeler tuş sayısına ve barındırdıkları karakter ölçülerine göre: **84, 101, 102 tuşlu**, harf dizinimlerine göre alfabetik klavyenin sol üstünde bulunan tuşa göre **Q klavye**, **F klavye** diye adlandırılırlar. **Q klavyeye QWERTY klavye** adı da verilir. Ülkemize ilişkin bir özellik de **Türkçe Karakter** bulunması özelliğidir. Türkiye için dilimize uygun klavye **F klavye**dir. Aşağıda **DOS 6.2**'nin yüklediği **F Klavye** örneği verilmiştir:



Türkçe (F) Klavye

3. YAZICI ve YAZICI ÇEŞİTLERİ

Yazıcı (Printer): İstenilen sonuçların, rapor ve dökümanların belge halinde çıkarılması gerektiği durumlarda, yazıcıdan yararlanarak istediğimiz çıktıyı kâğıt, asetat, aydinger vb. dönebiliriz. Yazıcıların önemi günümüzde kendini daha çok göstermektedir. Özellikle hız, bellek grafik ve çıktı kalitesi açısından incelenmektedir.



Nokta Vuruşlu Yazıcı

Yazıcı Çeşitleri:

a. Nokta Vuruşlu Yazıcılar (Dot Matrix Printer)

Bu tür yazıcılar, yazımı nokta modunda yazdıklarından **Nokta Vuruşlu Yazıcılar** olarak adlandırılır. Yazı ve grafikler noktalarından oluşur. Yazıcının kafasında bulunan iğneler aracılığı ile yazılır. İğne sayısı çoğaldıkta daha net çıktı elde edilir. 9, 24, 48 iğnelidir.

Bu yazıcılarda hız ölçüsü **CPS**'dir. (**ch**racters **p**er **s**econd: Saniyedeki Karakter Sayısı) Bu **360 CPS** hızındaki bir yazıcının saniyede **360 karakter** yazabileceğini gösterir. Yazım hızı, yazıcının normal modundaki hızdır; grafik ve çeşitli yazı fontlarında bu hız düşük olur.

Yazım kalitesi **LQ**, **NLQ** şeklinde tanımlanır. **LQ** (**L**etter **Q**uality) Mektup kalitesinde yazım, **NLQ** (**N**ead **L**etter **Q**uality) Mektup kalitesine yakın yazıdır.

b. Mürekkep Püskürtmeli Yazıcılar (Ink-Jet Printer)

Kağıt üzerine mürekkep püskürterek yazırlar. İğne yerine delikler vardır. Bu deliklerden mürekkep püskürtülerek yazım yapılır.

c. Satır Yazıcılar

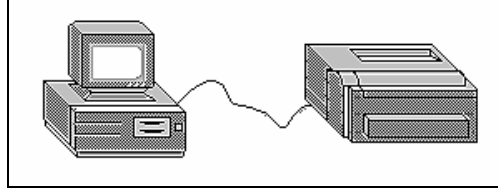
Bu tip yazıcılar, tüm satırı aynı anda basarlar. Bir satıra **132 karakter** yazabilecek ölçüde satır vuruşludurlar. Bir satır bittiğinde bir alt satıra geçerler.

d. Termal Yazıcılar

Termal yazıcılar, ısı teknolojisiyle çalışırlar. Mürekkebi eriterek kağıda basarlar. Net ve temiz çıktılar alınır.

e. Lazer Yazıcılar

En kaliteli çıktılarını aldığı yazıcılardır. Lazer teknolojisi ile çalışırlar. Bir sayfayı oluşturarak, sayfayı olduğu gibi basarlar. **300 dpi**, **600 dpi**, **1200 dpi** baskı kalitesinde basım yaptıklarından masaüstü yayıncılıkta kullanılırlar. Bu ölçüler çoğaldıkça daha net çıktı elde edilir. Harfler ve çizimler kırık çıkmaz. Basım hızları dakikada basabildiği sayfa ile ölçülür. Basım hızı yazıcının mekanik hızıdır.



Bilgisayar ve Lazer Yazıcı

Son zamanlarda Lazer teknolojisindeki gelişmeler, bu yazıcıların fiyatlarını ucuzlatmış ve daha çok kullanılabilir olmuştur.

Yukarıda saydığımız yazıcıların renkli basım yapabilme özellikleri de vardır. Renkli olması fiyatlarını etkiler.

4. DİSKET ve DİSKET ÇEŞİTLERİ

a. Disket

Bilgi depolanabilen, silinebilen esnek (**floppy**) manyetik plaktır. Bildiğimiz plaktan farkı, birbirini kesmeyen dairelerden oluşmasıdır. Daireler disketin ve kullanılan sürücünün yapısı ve kapasitesine göre **40, 80** adet olabilir. Biz bunlara **TRACK (İz)** adını veriyoruz. Bir de disketi enine kesen çizgiler vardır ki bunlar da her **Track**'da **9, 15, 18** ya da **36** tane bulunur. Bunlara da **SECTOR** adı verilir.

Bir disketi kullanabilmek için, bilgisayar tarafından bir program yardımı ile bu track ve sectorler açılır. Yani bilgisayar tarafından kullanılan işletim sistemi ve bilgisayara göre izler açılır. Bu olaya da **Formatlama** adı verilir.

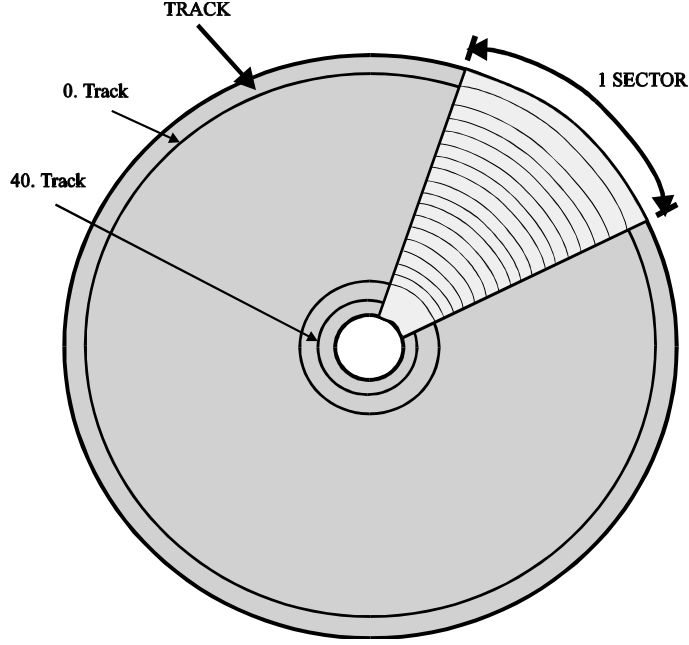
Disketler boylarına göre birkaç çeşittir. Bunlar **3.5"**, **5.25"**, **8"** gibi. (" İşareti inch anlamındadır.) Depolayabildikleri bilgi açısından ise **360 kb, 720 kb, 1.2 mb, 1.44 mb, 2.88 mb** kapasiteleri vardır. Bu terimler **KB=Kilo Byte**, **MB=Mega Byte** olarak tanımlanırlar. (İlerideki konularda bu kavramlar üzerinde duracağız.) Formatlanabilen ve kullanılan alanlar; bilgisayarın, sürücünün ve disketlerin özelliğine göre değişir.

Disket çeşitlerini şu şekilde sınıflandırabiliriz:

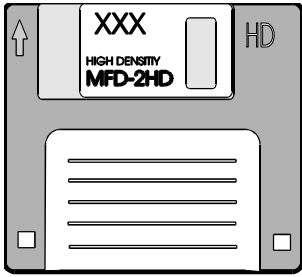
Boyut	Yoğunluk	Kapasite	İz	Sector
5.25 "	Double Sided Double Density	360 KB	40	9
5.25 "	Double Sided High Density	1.2 MB	80	15
3.5 "	Double Sided Double Density	720 KB	80	9
3.5 "	Double Sided High Density	1.44 MB	80	18
3.5 "	Extra Double Sided High Density	2.88 MB	80	36

Double Sided Double Density : Çift Yüzlü Çift Yoğunluklu

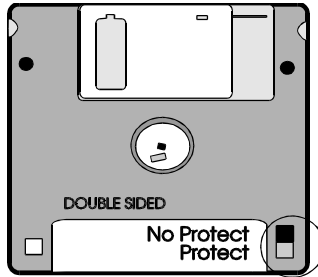
Double Sided High Density : Çift Yüzlü Yüksek Yoğunluklu



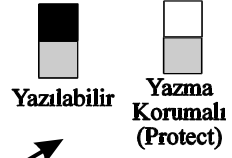
Disketin İç Yapısı



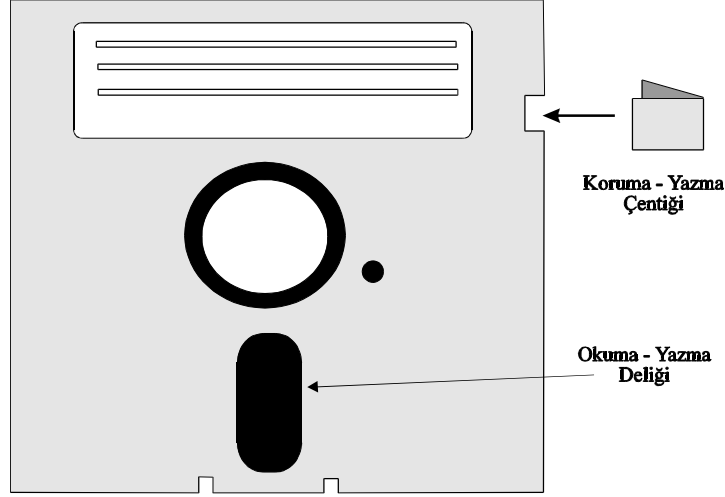
Önden Görünüş



Arkadan Görünüş

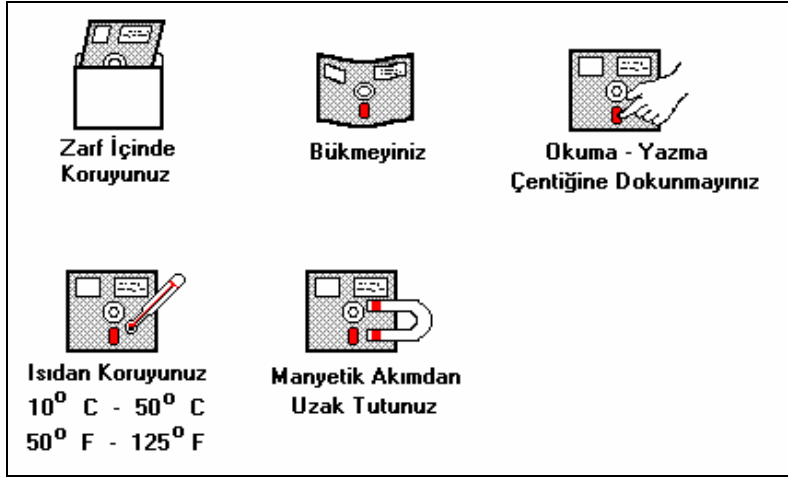


3.5 " lik Disket



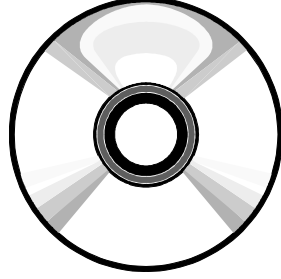
5.25 “ lik Disket

Disketlerimizi dış etkenlerden korumalıyız:



b. CD Disk

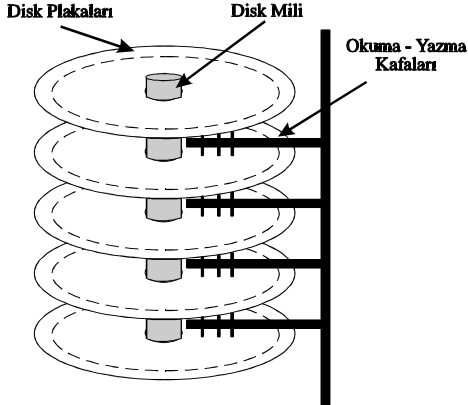
(Optik Disk) Lazer teknolojisi ile okuyup yazabilen disk türüdür. Normal diskete göre daha çok bilgi depolayabilir. Disket gibi kolay bozulmazlar.



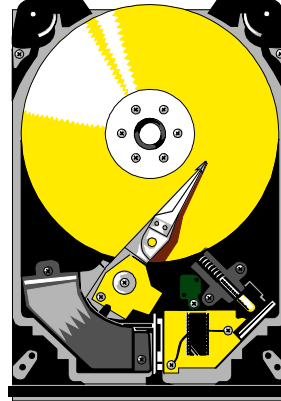
Optik Disk

c. Sabit Disk

Disk (Hard Disk): Çoklu plakanın, bir araya toplanmış biçimidir. Bunlar aynı anda kullanılabilirler. **120 MB, 245 MB, 340 MB** ve daha çok bilgi depolayabilen kapasitededirler. Birkaç yüz disketin depolayabileceği alana sahiptirler. **17 ms, 28 ms, 40 ms** (mili second) gibi hız kavramları vardır. **ms**'in anlamı mili saniyedir. Erişim hızını belirler. Disketten farklı ve avantajlı yanı, daha çok veri alabilmesi ve okunan bilginin hızlı ve sürekli kontrol edilmesidir.



Hard Diskin İç Yapısı



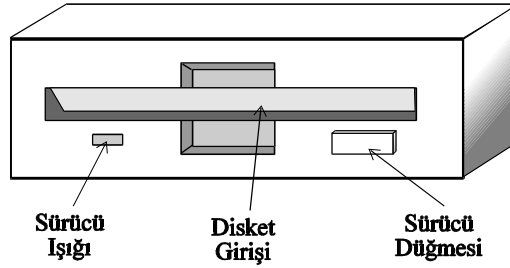
Hard Diskin Genel Görünüşü

5. DRIVER ve DRIVER ÇEŞİTLERİ

Driver (Sürücü): Disketin takıldığı ve okuma-yazma kafalarının bulunduğu yerdir. Disketler sürücüye takılarak çalışırlar. Sürücü sayesinde istenilen **iz** ya da **sector**'e, bilgiye erişilir.

a. 3.5 inch'lik sürücü

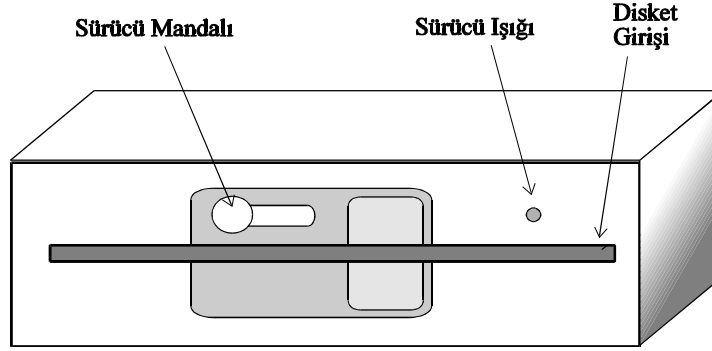
Adından da anlaşılacağı gibi **3.5 inch** boyutunda olan disketlerin takıldığı sürücüdür.



3.5 “ lik Disket Sürücü

b. 5.25 İnce'lik sürücü

5.25 inch boyutunda olan disketlerin takıldığı sürücüdür.



5.25 “ lik Disket Sürücü

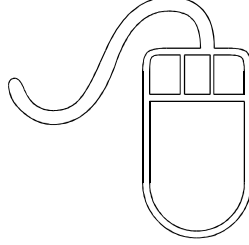
c. CD Sürücü

Lazer teknolojisiyle okuyup yazan CD disklerin takıldığı sürücüdür. Bu teknoloji ile bir CD diske daha çok bilgi aktarılabilir.

6. BİLGİSAYARDA KULLANILAN DİĞER YAN BİRİMLER

a. Mouse (Fare)

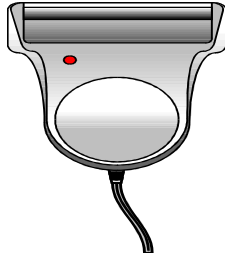
Bir girdi birimidir. Klavyenin kullanılmadığı, daha çok ekranda belirtilen bölümleri işaretleyerek çalışan, masa üzerinde gezdirilerek, ekranda hedeflenen bölüme gelince düğmeye basılarak çalışabilen aygıttır. Farenin altında bulunan yuvarlak bir top aracılığı ile imlecin ekranda, farenin sürüldüğü tarafa gitmesi sağlanır.



Mouse

b. Scanner (Tarayıcı)

Bir belge üzerindeki resim, yazı vb. bilgileri bilgisayara aktarma işlemini yaparlar. Belge üzerinde almak istediğimiz bölümü, tarayarak bilgisayara aktarır. **Hand Scanner** (El Tarayıcı) ve **Page Scanner** (Sayfa Tarayıcı) olarak iki çeşittir.

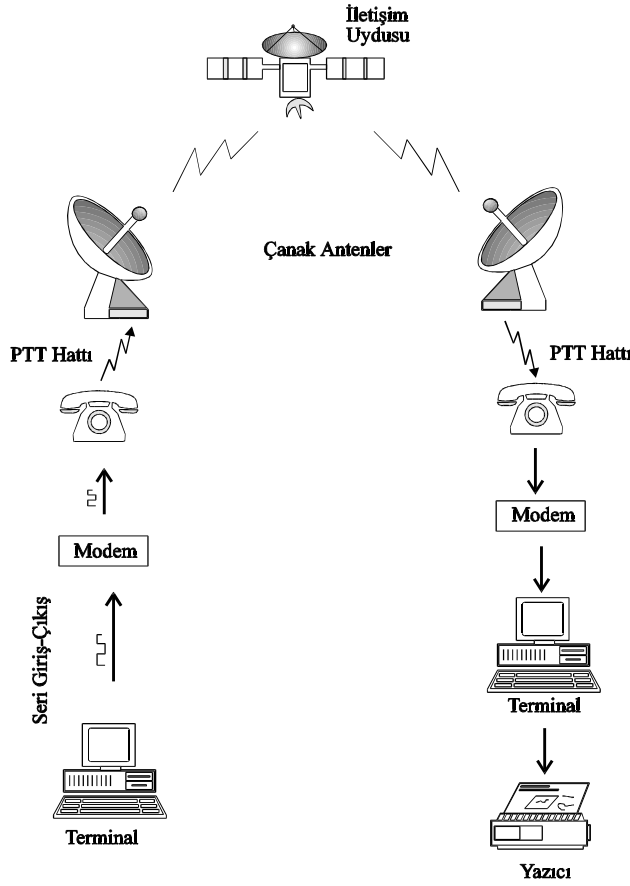


El Tarayıcı

c. Modem

Bilgisayarlar arasında, telefon aracılığıyla iletişimi sağlar. Bunun için bir modem kartı ve modem cihazı gerekir. Doğaldır ki iletişim yapılacak iki bilgisayarın da modemi bulunması gereklidir.

Günümüzde bilgisayarlar arasında iletişim gittikçe yaygınlaşmakta; bilgisayar bir fax-modem gibi kullanılabilir. Aşağıdaki şekil, bilgisayarlar arasında iletişimin nasıl kurulduğunu göstermektedir.



Bilgisayarlar Arasında İletişim

d. Plotter (Çizici)

Daha çok mühendislikte çizim alanında kullanılır. Renkli çizim yapabilme özelliğine de sahiptir. Renkli çizimde, birden çok kalem ile çizim sağlanır.

e. Oyun Çubuğu (Joystick)

Günümüzde bilgisayarlar yalnız iş alanında kullanılmakla kalmayıp, zeka oyunları için de kullanılmaktadır. Daha çok eğitim amacıyla hazırlanan bu zeka oyunları, çocuğun pratik zekasını, karar verebilme, refleks ve el çabukluğunu da artırmaktadır. Bu oyunların oynanabilmesi için, bazı durumlarda klavye tuşlarıyla noktaların yönetimi olanaksızlaşır. Bu nedenle, noktaların hareketi, ileriye-geriye, sağa-sola doğru hareket edebilen çubuk aracılığıyla sağlanır. Bu çubuğa oyun çubuğu (Joystick) denir.

f. Işık Kalem (Light Pen)

Grafik ve çizim için kullanılan bir giriş birimidir. Ekran üzerine kullandığımız programda bulunmayan bir şekil çizmek ve karakter seti dışında bir çizim oluşturmak istediğimizde ışıklı kalem'den yararlanırız. Bir kağıda çizer gibi ekran üzerine çizim yapabiliriz.

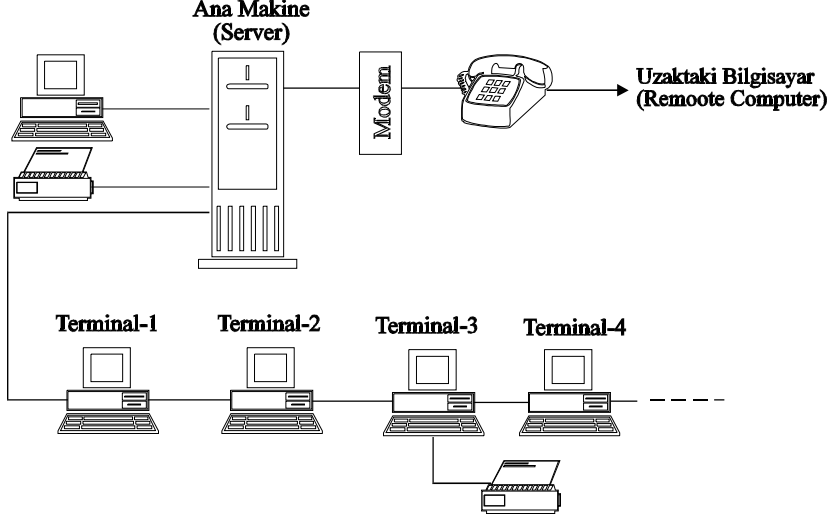
g. Teyp

Başka bir dış bellek aracıdır. Bildiğimiz teyp kasetleri ve buna benzer şekillerde olur. Bu tür bellek biriminde bilgileri almak zaman alıcıdır. Erişmek istediğimiz bilgiyi alabilmek için, o bilgiye kadar tüm bilgiler okunur.

7. BİLGİSAYARLAR ARASINDA BAĞLANTI

Bilgisayarlar birbirlerine bağlanarak çalışabilirler. Bu tür sistemlere **network** sistemi denir. Kullandıkları işletim sistemi farklıdır. Bu tür işletim sistemi DOS komutlarının çoğunu kullanabilirler.

Network bağlantılı bilgisayarlar biribiri ile alışveriş yapabilirler ve aynı anda bir kaç kişi tarafından kullanılırlar. Aşağıda bir network bağlantısının şemasını görüyorsunuz:



Bilgisayarları birbirine bağlamak için **Ethernet Kartı** takılır ve uygun bir işletim sistemi kurulur.

SORULAR

1. Giriş birimleri nelerdir? Sayınız.
2. Kontrol Birimi'nin görevi nedir?
3. Bellek Nedir? Kaçaya ayrılır? Sayınız.
4. Disket ve Hard Disk arasındaki farklar nelerdir?
5. Çıkış birimleri nelerdir? Sayınız.
6. Pixel nedir? Tanımlayınız.
7. Yazıcı çeşitlerini sayınız ve çalışma sistemlerini açıklayınız.

BÖLÜM III.

BİLGİSAYARIN ÇALIŞMA İLKESİ

Bir bilgisayarın çalışması için **İşletim Sistemi**'ne gereksinim vardır. Bazı bilgisayarlarda işletim sistemi, bilgisayarın belleğindedir. Üzerinde işletim sistemi olmayan bilgisayara hangi işletim sistemi ile çalışılacaksa, disketten yüklenir ve bilgisayar verilen komutları algılamaya hazır duruma getirilir. Bu işlemden sonra kullanacağımız program, bilgisayarın belleğine yüklenir. Bu, bir paket program ya da çalışılması istenen yüksek düzeyde bir dil olabilir.

Bilgisayarların çalışma sistemleri **Giriş-İşlem-Çıktı** sırası ile dir.

Giriş: Programla ilgili verilerin uygun giriş birimleri (klavye, mouse, ışıklı kalem, disk/disket vb.) ile bilgisayara aktarılmasıdır.

İşlem: Yüklenen veriler programın işleyiş ve akışı doğrultusunda **Merkezi İşlem Birimi**'nde mantık süzgecinden geçirilerek işlenirler. Bilgisayara girilen bilgiler, bilgisayar tarafından işlem biriminde kontrol edilerek onaylandıktan sonra verilen komutlar doğrultusunda işlenirler.

Çıktı: Elde edilen sonuçlar, uygun çıktı birimlerinden (ekran, yazıcı, disk/disket vb.) alınır.

A. DİSKETİN SÜRÜCÜYE TAKILMASI

Bilgisayarı açabilmek için işletim sistemi disketi, yani **MS-DOS** disketi gereklidir. Bu disketinizi hazırlayınız. İçinde Harddisk bulunan bilgisayarlarda, sistem genellikle yüklüdür. Bu durumda, disket bilgisayara takılmadan açılır.

Disketi, baş parmağınız ile diğer parmaklarınız arasında etiket kısmından tutunuz. Etiket üste gelecek biçimde ve yerine oturacak şekilde sürücüye takınız. Sürücünün dışındaki mandalı kapatınız.

B. SİSTEMİN AÇILMASI

Disketinizi sürücü gözüne taktıktan sonra elektrik anahtarını açınız. Birazdan bip sesi gelerek, **CPU, ROM, RAM** ve **KLAVYE** vb. ünitelerin bağlanıp bağlanmadığını kontrol eden otomatik testi yapacaktır. Test sonucunda hata yoksa sistem disketten yüklenerek açılma işlemi yapılmış olacaktır.

Genelde sistemin açılışı, sırasıyla şu biçimde olur:

Bu işlem bilgisayardan bilgisayara değişiklik gösterir. Bunun nedeni de **AUTOEXEC.BAT** dosyasıyla istenilen biçimde açmayı sağlamaktadır.

Bilgisayar her açılışta sistem kartı, genişleme kartı üzerindeki **RAM** belleği kontrol eder. Bu işlemler sırasında bir hataya rastlanılırsa sistem sizi uyaracaktır.

Sistem bu şekilde açıldıktan sonra ikinci bir bip sesi gelecek, ve aşağıdaki mesajlar çıkacaktır.

```
Starting MS-DOS...

Current date is Tue 1-01-1980
Enter new date (mm-dd-yy)  ↵
Current time is 0:00:29.87
Enter new time: _  ↵

Microsoft(R) MS-DOS(R)  Version x.xx
                (C) Copyright Microsoft Corp 1981-1993

A>
```

Biz burada istersek tarih ve zamanı değiştirebiliriz. Şöyle ki:

Enter new date (mm-dd-yy)_ mesajı çıkacaktır. Bu **Yeni Tarihi Giriniz** anlamındadır. Mesajın yanında, burada olduğu gibi "_" işareti çıkacaktır. Biz buna **imleç** (kursör) diyoruz. Eğer tarihi değiştirmek istiyorsanız aşağıdaki biçimde yeni tarihi giriniz ve **Enter** tuşuna basınız.

```
Enter new date (mm-dd-yy) _
mm-dd-yy ya da mm/dd/yy
ya da
aa-gg-yy ya da aa/gg/yy
```

aa=Ay - gg=Gün - yy=Yıl

Örnek:

8-18-1994 ya da **8/18/1994**

5-19-1919 ya da **5/19/1919** şeklinde girilir.

BÖLÜM IV. İŞLETİM SİSTEMİ

A. GİRİŞ

İşletim Sistemi: Bilgisayarın açılışından kapanışına kadar, kullanıcı ile bilgisayar arasındaki bağlantıyı sağlayan, komutları algılamaya, programları ve çevre ünitelerini kullanılır duruma getiren, sistem yazılım programlarıdır.

İşletim sistemi, giriş birimlerinden aldığı komutlara göre bellek ile giriş-çıkış birimleri arasında alışverişi, programların çalıştırılmasını ve dosya işlemlerinin gerçekleştirilmesini sağlayan ve denetleyen programdır. Kısaca işletim sistemi; bilgisayarın can damarıdır.

Bilgisayarların tiplerine, modellerine göre kullandıkları işletim sistemi farklıdır.

B. İŞLETİM SİSTEMİ ÇEŞİTLERİ

- MS-DOS ve PC-DOS
- OS/2
- CP/M
- UNIX
- XENIX
- WINDOWS-NT

1) MS-DOS : **M**icro**S**oft**D**isk **O**perating **S**ystem'in kısaltılmışıdır. Microsoft Disk İşletim Sistemi anlamındadır. En çok kullanılan sistemdir. Biz de kitabımızda bu sistem üzerinde duracağız. **MS-DOS** Microsoft Corporation tarafından gerçekleştirilmiştir. 16 bit'lik mikrobilgisayarlar için yazılmıştır. PC-DOS, MS-DOS'un IBM uyarlamasıdır.

2) UNIX : Çok kullanıcıli bilgisayarlar için geliştirilmiş işletim sistemleridir. Kullanılışları tek kullanıcıli bilgisayarlara göre biraz karmaşıktır. Aynı anda, aynı bilgi deposuna birkaç terminalden giriş yapılabilir.

3) WINDOWS-NT : Pencereli işletim sistemi de diyebiliriz. WINDOWS-NT'nin avantajı aynı anda birden çok dosyayı açabilmek ve dosyalar arasında kolay geçiş yaparak alışveriş yapabilmektir.

C. DOS İŞLETİM SİSTEMİNE GİRİŞ

Piyasada en çok kullanılan bu işletim sisteminin kullanımı hem basit hem de pratiktir. MS-DOS İşletim Sisteminin birkaç uyarlaması (versiyon) vardır. Bunlar ... 4.01, 5.00, 6.00, 6.20 gibi... Aralarında çok fark olmamasına rağmen son versiyonlar daha çok işleve sahiptirler.

Biz, işletim sistemi konusunda bir kullanıcı (operatör) için gerekli olan veya olabilecek bilgi ve komutları göreceğiz. Bu bilgiler ve komutlar ışığında istediğimiz çalışmayı yapabilir ya da ileride bu bilgilerimizi genişletebiliriz.

D. DOS İŞLETİM SİSTEMİ KAVRAMLARI

MS-DOS işletim sistemini öğrenir ve kullanırken bazı kavramlarla karşılaşacağız. Bu kavramların anlamlarını bilmemiz, DOS'u daha başlarken kullanmamızı kolaylaştırır.

1. SÜRÜCÜ ADI

MS-DOS İşletim Sistemini kullandığımızda karşımıza **A>** işareti çıkar ki bu da MS-DOS'un prompt'udur. Bu promptu gördüğümüz andan itibaren bilgisayarımız MS-DOS komutlarını almaya hazırdır. Burada **A>** işareti sürücünün adını belirlemektedir. Tek disket sürücülü bilgisayarlarda, sürücü **A>**'dir. Çift disket sürücülü bilgisayarlarda ikinci sürücü **B>**'dir. İçinde disk ünitesi bulunan bilgisayarlarda, yani Hard Disk **C>** promptu ile gösterilir.

Bilgisayarımız birden fazla sürücülü veya Hard Diskli ise çalışırken, bir sürücüden diğerine geçmek gerekir. Bu geçisi aşağıdaki şekilde gerçekleştirebiliriz:

A>B:	↵	B sürücüsüne geçer.
B>C:	↵	C sürücüsüne geçer.
C>A:	↵	A sürücüsüne geçer.

Hangi sürücüde çalışıyorsak, karşımızda o sürücünün promptu vardır. Sürücü değiştirmek için, geçmek istediğimiz sürücünün harfini ve yanına (:) **iki nokta** işaretini yazarak (↵) ENTER'e basmamız yeterlidir.

2. YOL ADI (PATH NAME)

Yol adı olarak tanımlanır. Yol adı belirlenirken sürücü, dizin (directory) belirtilir. Bir dosya ile ilgili işlemlerimizi yapabilmemiz için, o dosyayı bulmamız gerekir. Bunun için de dosyanın bulunduğu sürücü ve dizine girmemiz gerekir ki; bu da dosyanın bulunduğu yerin belirtilmesiyle olur.

MS-DOS'da komutların yazılışı sırasında yol adı vereceğiz. Yazılım kalıbı aşağıda verildiği gibidir:

sürücü:\dizinadı\dizinadı\...\dosyaadı.uzantısı

Yol belirleme komutu **PATH**'dir. Eğer **PATH** kullanılmıyacaksa, Yukarıda verdiğimiz örneği diğer komutlarla sürücü, dizin ve dosya adını yazıp yol belirleyerek kullanabiliriz.

3. DİZİN (DIRECTORY)

Dizin: Disket ya da diske kaydedilen bilgilerin daha kolay bulunabilmesi için, kullanacağımız dosyalar topluluğudur. Birbirine benzeyen ve ilgilendiren dosyaları bir dizin içerisinde toplarsak hem çalışmamız, hem de aradığımız dosyayı bulmamız kolaylaşır.

Bir muhasebe bürosunda bulunan dolapları ele alacak olursak; her dolapta belirli defterler vardır. Defterler karışıklığı önlemek için sınıflara göre ayrılmıştır. 1. dolapta Birinci Sınıf Defterler, 2. dolapta İkinci Sınıf Defterler, 3. dolapta Kooperatif Defterleri bulunduğunu varsayalım. İşte bu dolaplara **dizin (directory)** diyebiliriz. Bu büroda 3 dizin vardır. Her dolabın içinde de o dolabın sınıfına giren mükelleflere ait dosyalara da **dosya**, (kütük, file) adını verebiliriz. Bu dosyaların üzerinde de ait olduğu mükelleflerin adları karışıklığı önlemek ve kolay bulmak için yazılıdır. Bu oda gibi birkaç oda daha varsa, o zaman odalara dizin, dolaplara da alt dizin denir.

a. ANA DİZİN (ROOT DIRECTORY)

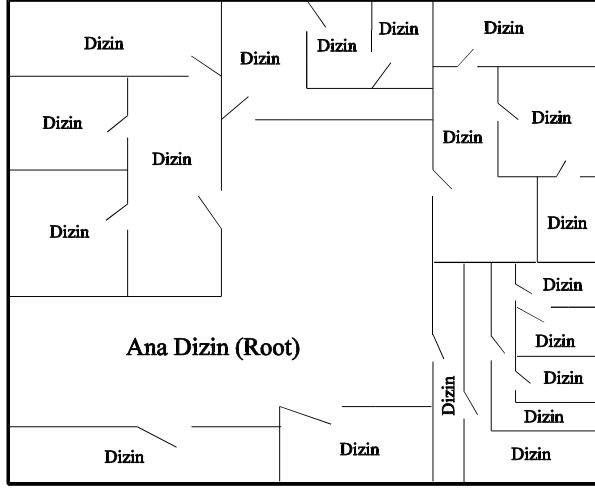
Diskette kayıtlı bilgiler daha çok dizinler altında saklanır. Bunun nedeni de birbiri ile ilgili dosyaları bir araya toplayarak, arandığında kolay bulunmasıdır. Disketi ilk açtığımızda ana dizindeyiz demektir. Yukarıdaki örneğimizi ele alacak olursak büroya da **Ana Dizin** adını verebiliriz. Bir başka örnek verecek olursak; bir siteye ilk girdiğimizde, önce cadde ya da sokağa girmeye benzetebiliriz.

Ana dizine **ROOT Directory** adı verilir.

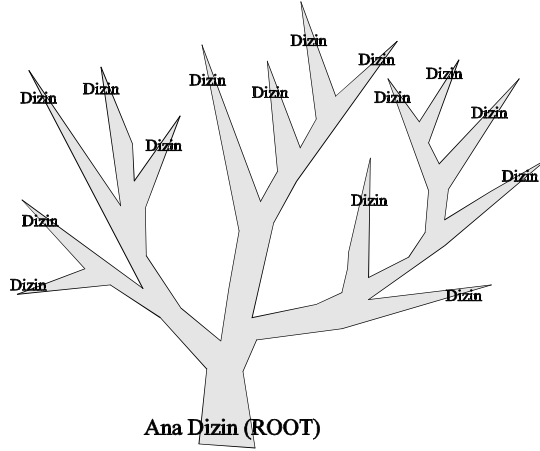
b. ALT DİZİN (SUB DIRECTORY)

Ana dizinin altında bulunan dizinler alt dizinlerdir. Her alt dizinin altında birden çok alt dizin bulunabilir. Alt dizinleri de sitedeki bloklara, bloklardaki apartmanları sitenin alt dizini, daireleri de apartmanın alt dizini, odaları dairenin alt dizini olarak düşünecek olursak dizin kavramını daha kolay anlayabiliriz.

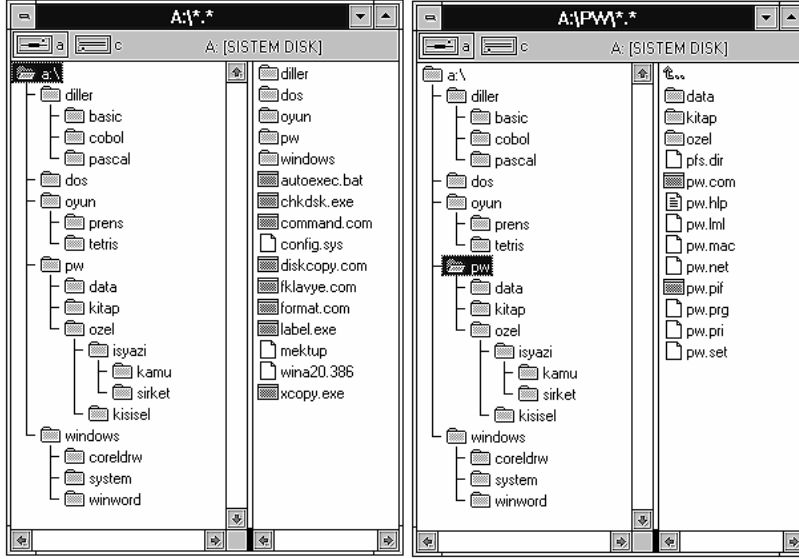
Aşağıda vereceğimiz iki şekil dizini değişik yönlerden sunmaktadır. 1. şekilde bir bina örneği ve giriş Ana Dizin, odalar ise alt dizinlerdir. 2. şekilde ise bir ağaç üzerinde dizin şeması verilmeye çalışılmıştır. Burada da kök Ana Dizin, dallar da alt dizinlerdir.



Dizin Şeması 1



Dizin Şeması 2



Ana Dizin Şeması

Alt Dizin Şeması

Bu iki şeklin sol taraflarında dizinler ağaç şeklinde tamamen verilmekte, şekillerin sağ taraflarında ise kendi içinde olan dizin ve dosyalar verilmekte. Soldaki şekilde Ana Dizin, sağdakinde ise PW dizini verilmekte.

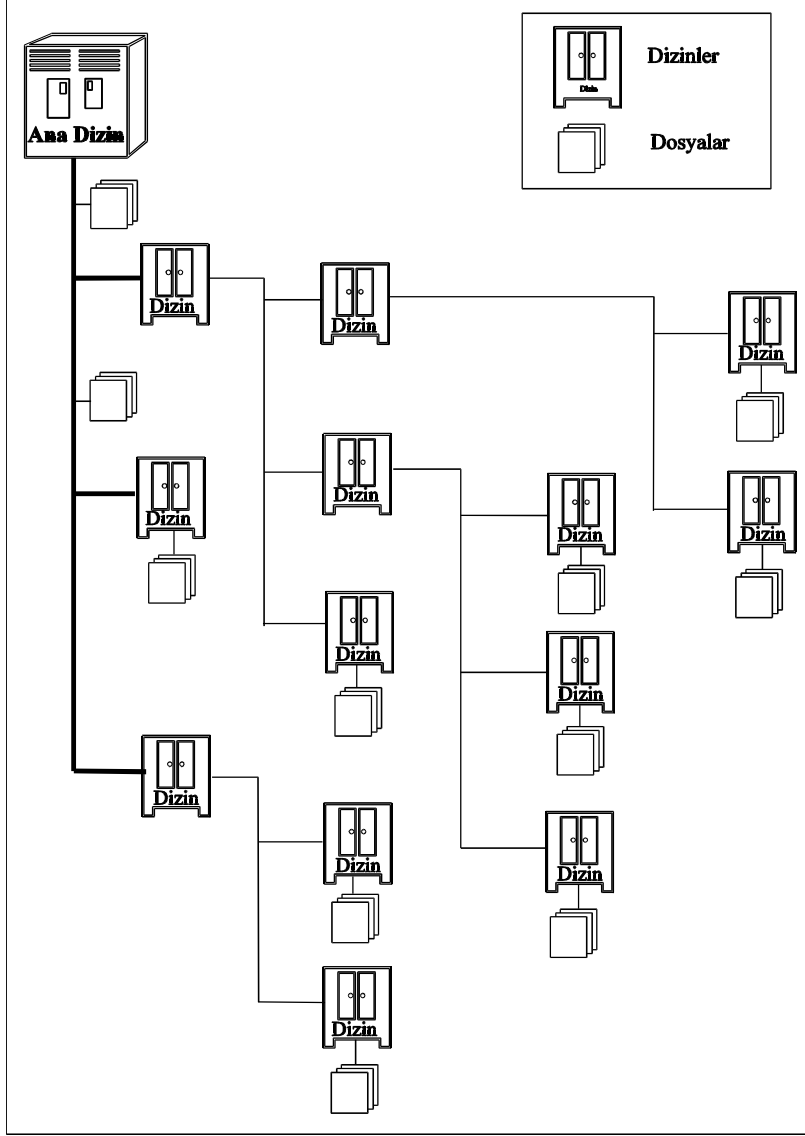
4. DOSYA (KÜTÜK, FILE)

Bilgisayarda birbiri ile ilgili bilgileri içeren (mektup örneği, program, veri grubu, yasalar ve tüzükler gibi) bilgi grubuna **dosya** denir.

Biz bu bilgileri bilgisayarın yan belleği olan Disk, Disket, Teyp üzerine kaydederek, gerektiği zaman kullanılmak üzere saklarız. Bu bilgileri bu ünitelere kaydederken bunlara bir isim veririz. Aradığımız zaman bu isimle çağırırız.

Bilgisayarlarda dosya organizasyonu üç biçimde olur:

1. Sıralı Erişimli Dosyalar.
(SEQUENTIAL ACCESS FILES)
2. Doğrudan (Rastgele) Erişimli Dosyalar.
(RANDOM ACCESS FILES)
3. Index Dosyalar.
(INDEX SEQUENTIAL FILES)



Dizin ve Dosya Şeması

a. DOSYA ADI VE KURALLARI

Dosya adı verirken MS-DOS tarafından istenen kurallar vardır. Bu kurallara uymak zorundayız.

Dosya adı verirken şunlara dikkat etmeliyiz:

1. Dosya adı **8 karakteri** geçemez. Dosya adına ek olarak uzantı kullanılabilir. Bu durumda uzantı **3 karakteri** geçmemelidir. Dosya adından sonra **nokta** (.) konulup uzantı yazılmalıdır. **VERGLIAD, MUSTERI.BAS, BORDRO.EXE** gibi...

<u>Dosya Adı</u>	<u>Uzn</u>
8 Karakter	3 Karakter

2. Dosya adı verirken özel karakterler (" . , / \ ; : = < > [] + gibi işaretler) kullanılmamalıdır. **SINIF/3, CARI+HES** yazmak yanlıştır.

3. Dosya adı verirken Türkçe karakterler kullanılmamalıdır. (**ş,ç,ı,ö,ğ,ü** gibi...) Bu karakterler bilgisayar için özel karakterdir. **MS-DOS**'un **6.20** versiyonunu kullanıyorsanız ve **Türkçe Karakter Setini** yüklediyseniz, **büyük İ** ve **Küçük ı** dışındaki Türkçe karakterleri de kullanabilirsiniz.

4. Dosya adı verirken bilgisayarda kullanılan komut ve deyimler kullanılmamalıdır. **END, COPY, FILES** gibi...

5. Bir dizin içerisinde daha önce kullanılan dosya adı ve uzantısı aynen yazılamaz. Böyle bir ad verilecek olursa, daha önce bulunan dosyanın içeriği silinerek yeni dosyanın içeriği yer alacaktır. Böyle bir durumda bazı programlar aynı dosyanın üzerine yazılıp yazılmayacağını uyaracaktır.

6. Dosya adı verirken programın işlevine ve içeriğine uygun bir ad verilmesi hem karışıklığı önler, hem de aradığımız zaman konuyla ilgili isimde olduğu için kolayca bulabiliriz. Bir bordro programı yapıyorsak **BORDRO**, stok kontrol programı yapıyorsak **STOK** gibi adlar vermek işimizi kolaylaştırır.

Dosya adları küçük harfle verirse de, bilgisayar tarafından büyük harfe çevrilir.

b. DOSYA UZANTISI

Dosya uzantılarının bilgisayar için bazı anlamları vardır. Bunlar:

<u>UZANTI</u>	<u>ANLAMI</u>
?????????. ASM	ASSEMBLER kaynak programları
?????????. BAK	BACKUP (yedek) dosyası
?????????. BAS	BASIC programları
?????????. BAT	BATCH (Otomatik) komut dosyaları
?????????. COB	COBOL kaynak programları
?????????. COM	Doğrudan çalıştırılabilen dosya
?????????. CRF	Geçici kaynak dosyaları
?????????. DAT	DATA (Veri) dosyası
?????????. EXE	Makine dilinde dosyalar
?????????. FOR	FORTRAN kaynak programları
?????????. KEY	Anahtar dosyası
?????????. LST	Derleyici yazı listeleri
?????????. NDX	İndexlenmiş dosya
?????????. OBJ	Amaç kod modülleri
?????????. OVR	Çakışma modülleri
?????????. PAS	PASCAL kaynak dosyaları
?????????. PL1	PL/1 program dosyası
?????????. PRG	dBASE program dosyası
?????????. \$\$\$	Geçici sistem dosyaları

Uzantıları **EXE**, **COM** ve **BAT** olan dosyalar, işletim sisteminde direk olarak çalışırlar.

5. DOSYA ADLARINDA KULLANILAN JOKER KARAKTERLER

İşletim sisteminde Dosya Adı yazılırken, bazı kolaylıklardan yararlanmak için Joker işaretler dediğimiz işaretleri de kullanacağız. Bunları, birden çok dosya ile ilgili işlem yaparken zaman ve emekten kazanma açısından kullanmamız gerekir. Bunlar:

- * Yıldız işareti tüm karakterleri simgeler.
- MU*.*** Adı **MU** ile başlayan, uzantısı ne olursa olsun bütün dosyaları tanımlar.
- *.*** Adı ve Uzantısı ne olursa olsun bütün dosyaları tanımlar.
- *.BAT** Adı ne olursa olsun, uzantısı **BAT** olan dosyaları tanımlar.
- MU*.D*** Adı **MU** ile başlayan, uzantısı **D** ile başlayan bütün dosyaları tanımlar.
- ? Soru işareti bir karakterin yerine geçer.
- BO??.*** Adı **BO** ile başlayan ve **4 karakter**li olan, uzantısı ne olursa olsun bütün dosyaları tanımlar.
- M?????.DAT** Adı **M** ile başlayan ve **6 karakter**li olan, uzantısı **DAT** olan dosyaları tanımlar.
- MU*.*??** Adı **MU** ile başlayan ve uzantısı **2 karakter**li olan dosyaları tanımlar.
- WIN?????.E*** Adı **WIN** ile başlayan **7 karakter**li olan, uzantısı **E** ile başlayan dosyaları tanımlar.

6. İÇ VE DIŞ KOMUTLAR

MS-DOS işletim sisteminde iki tür komut vardır. Bu komutlar **İç** ve **Dış komut** olarak tanımlanırlar.

İç Komut: İşletim sistemi aracılığı ile yüklenen bilgisayarın belleğinde olan komutlardır. Bu türden komutları kullanırken MS-DOS disketinin sürücüde takılı olup olmaması önemli değildir.

Dış Komut: Bu türden komutların uygulanabilmesi için MS-DOS disketinin kullanılan sürücüde olması gerekir.

MS-DOS komutlarına geçmeden önce kullanacağımız işaret ve deyimlerin neyi tanımladığını görelim:

↵ veya **<Enter>**: Bu işaret klavyedeki **ENTER** tuşunu tanımlar. Bu işaretin olduğu yerde **ENTER** tuşuna basmamız gerekir. Burada **ENTER** tuşunun görevini tanımlamamız gerekiyor: Enter, bilgisayarın belleğine herhangi bir bilgiyi göndermeye yarar. Girilen bütün bilgileri **ENTER** tuşu aracılığıyla bilgisayarın belleğine göndeririz. Bu işaret yerine **ENTER** yazıyorsa yine **ENTER** tuşuna basmamız gerekir.

<.....> : Bu işaretler içinde tuşlar verilmiştir. Bu işaretler ve içindekiler yazılmayacak , yerine içinde yazılan tuşa basılacaktır. Örnek: <Enter> yazılı ise, Enter tuşuna basılacaktır.

[] : Köşeli parantezlerin içindekileri yazma zorunluğu yoktur. Yapılacak işe göre değişiklik gösterir.

(..) : Nokta nokta ile gösterilen seçenekler isteğe bağlı olarak yinelenabilir.

Köşeli Parantez [] ve < > işaretleri yazılmayacaktır. Bu işaretlerin dışında verilen, işaret ve sembolleri yazmak zorunludur. Bunlar Virgül (,), Noktalı Virgül (;), Eşit İşareti (=), Üst üste iki nokta (:), Soru İşareti (?), Slay (/), Ters Slay (\) işaretleri gibi...

Komutlar, büyük ya da küçük harfle girilebilir. Küçük ya da büyük harf olması fark etmeyecektir. Bilgisayar bunları kendiliğinden büyük harfe çevirecektir.

Not: Çalışmaya başlamadan önce, şunu kesinlikle unutmamalıyız: Yapacağımız herhangi bir harf veya yazım hatası ya istediğimiz sonucu vermeyecek ya da bilgisayar tarafından kabul edilmeyerek hata mesajı ile uyarılacaktır.

E. İŞLETİM SİSTEMİNDE KLAVYE KULLANIMI

Klavye: Bilgisayarın çalışabilmesi için gerekli olan ve en çok kullanılan girdi birimidir. Bilgisayara gerekli verilerin girilebilmesi için harf, rakam vb. kullanılan tuşlar topluluğudur.

Bilgisayara girilen karakterler, işlenmesi için **MİB**'e (Merkezi İşlem Birimi) gönderildikleri an ekranda da gösterilir.

Klavye 4 kısımdan oluşur:

1. Daktilo (Alfasayısal) Tuşları
2. Edit Tuşları
3. Sayısal Tuşlar
4. Fonksiyon Tuşları



Klavye

1. Daktilo (Alfasayısal) Tuşları:

Bu tuşların çoğu iki ya da üç tuşla birlikte kullanılırlar.

Alfabe ve sayıların bulunduğu tuşlar bölümüdür. A'dan Z'ye kadar olan harfleri, 0'dan 9'a kadar olan rakamlar ve özel karakterleri içermektedir:

(! @ # \$ % ^ & * () _ - + = { } [] ; : " ' , . < > / ? \ |)



Daktilo (Alfasayısal) Tuşları

Yukarıda verdiğimiz tuşları yazmak için karakterini gördüğümüz tuşa basıldığında, o karakter monitörde belirecektir. Bu olaya basılan tuşun "**yankılanması**" adı verilmektedir.

Not: Aşağıda klavyede bulunan tuşların genel kullanımını vereceğiz. Bazı uygulama programlarında kullanımı değişiklik gösterebilir. Bu nedenle örneğin DOS'da kullanırken bile burada yazılanı yerine getirmeyebilir. Çeşitli kelime işlem programlarında değişik biçimde kullanılabilir.

Bu karakterlerin bazılarını, bir tuşa tek başına değil de ikinci bir tuşu da kullanarak yazmak zorunda da kalabiliriz. Tuşlarla ilgili konumuzda + işareti o tuşlara birlikte basılacağını göstermektedir.

Bu tuşların görevlerini ve kullanılışlarını sırasıyla görelim:

Daktilo Tuşları kısmında bulunan özel amaçlı tuşlar:



Enter tuşu, verilen komut ve bilgileri belleğe göndermede ve bir alt satıra geçmede kullanılır. **RETURN** adıyla da bilinmektedir. ↵ işareti ile gösterilir. Her işlem sonunda bu tuşa basmamız gerekmektedir. Bu tuşa basılmadan komutlar ve bilgiler bilgisayar tarafından belleğe gönderilmezler.



Değiştirme Tuşu

Bu tuşla birlikte herhangi bir harf tuşuna basıldığında, o tuşun büyük harfi yazılır. Eğer **Büyük Harf Kilitleme Tuşu** (CapsLock) basılı ise bu kez tersini yaparak o harfin küçüküğünü yazacaktır. Aynı zamanda bir tuşun üzerinde iki karakter varsa, Shift tuşu ile birlikte o tuşa basıldığında bu kez de, o tuşun üzerindeki karakteri yazacaktır.

Örnek:

- Shift+g** Büyük **G** yazacaktır.
- Shift+G** Küçük **g** yazacaktır. (Caps Lock açık ise)
- Shift+I** Bu tuşun üzerinde iki karakter **I** ve **!** işaretleri olduğundan, üstte ! işareti olduğu için, **!** işaretini yazacaktır.

Shift Tuşunun Özel Görevleri:

Shift+PrtSc (Print Screen) Ekrandaki görüntüyü aynen yazıcıdan verir.



Kontrol Tuşu

Bir başka değiştirme tuşu da **Ctrl** ile gösterilen kontrol tuşudur. Bu tuş birçok işlevi yerine getirebilecek şekilde programlanmıştır. ^ işareti ile de gösterilir. ^C, ^S gibi.

Control Tuşunun Görevleri Şunlardır:

Ctrl+Alt+Del	Bilgisayarın açma kapama düğmesine dokunmadan, bilgisayarın belleğini temizleyerek kapatır ve yeniden açar. Biz buna RESET adını veriyoruz. Program kilitlenmeleri gibi durumlarda kullanılır.
Ctrl+C ve Ctrl+Break	Komut iptal etmeye yarar. Çalışmakta olan programı durdurur.
Ctrl+S	Bir dosya taraması sırasında ekrandaki sayfa akışını durdurur. Herhangi bir tuşa basılınca sayfa akışı sürer. Ctrl+NumLock tuşları da aynı sonucu yaratır.
Ctrl+Num Lock	Ekrandaki görüntü kaymasını durdurur. Herhangi bir tuşa basarak sayfanın devamı görülebilir.
Ctrl+P ve Ctrl+PrtSc	Ekrandaki görüntünün ve bellekte bulunan komutların yazıcıdan alınmasını sağlar.



Alternatif Tuşu

Alt tuşu da bir değiştirme tuşudur. Bu tuşla birlikte 0'dan 255'e kadar bilgisayarda bulunan karakterlerin yazılması sağlanır.

Alt+154	154'e karşılık gelen Ü karakterini,
Alt+171	171'in karşılığı ½ karakterini,
Alt+246	246'in karşılığı ÷ karakterini vb.



Bir diğer alternatif tuşu da **Alt Gr** tuşudur. Bu tuş, bir tuşun üzerinde ikiden çok işaret varsa bu işareti yazmak için kullanılır. Yani bir tuşun üzerindeki üçüncü işareti basar.



Büyük Harf Kilitleme Tuşu

Caps Lock, sürekli büyük harfte yazmak istenildiğinde, bu tuş kullanılır. Bu tuşa bir kez basıldığında bu tuşun üzerinde bulunan ışıklı gösterge yanacaktır. Bundan sonra klavye büyük harfe kilitlenir ve büyük harf yazar. Bir kez daha basıldığında, eski konumuna döner.



Geri Dönüş Tuşu

Esc tuşu çeşitli uygulama programları tarafından değişik biçimlerde tanımlanmaktadır. Programcılar tarafından programdan çıkış ve geri dönüş tuşu olarak tanımlanırlar. Yürütmeyi durdurma ve komutu iptal etme özelliği vardır.



Ara (Boşluk) Tuşu

Space tuşu, klavyenin altında yer alan uzun çubuk görünümlü tuştur. Bu tuş sözcükler arasında veya gereken boşlukları vermek için kullanılır.

Bu tuşa basılmadan yazılan şöyle bir yazı;

BilgisayarÖğrenmekÇokKolay

Bu tuşa basılarak yazıldığında;

Bilgisayar <Space> Öğrenmek <Space> Çok <Space> Kolay

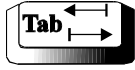
Bilgisayar Öğrenmek Çok Kolay şeklinde olacaktır.



Geri Silme Tuşu

Back Space tuşuna, her basışta imleç, bir karakter geri gider ve her geri gidişte yanında bulunduğu karakteri siler.

Bilgisaray_ <Back Space>
Bilgisara_ <Back Space>
Bilgisar_ <Back Space>
Bilgisa_



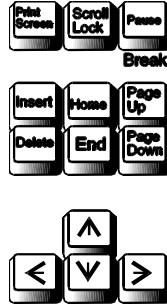
Durak Tuşu

Tab tuşuna her basışta imleç, **8 karakter** ileriye gider. Bu tuşu çizelge yaparken, durak yapmak amacıyla kullanırız.

Shift+Tab Her basışta imleç **8 karakter** geri gider.

2. Edit Tuşları

Bu tuşlar düzeltme tuşlarıdır. Ekran veya satır editörünü kullanmamız gerektiği durumlarda, düzeltme yapmak amacıyla bu tuşlar kullanılır.



Edit Tuşları



Ekranı Yazdırma Tuşu

Print Screen tuşu, **Shift tuşu** ile birlikte basılırsa ekrandaki görüntüler, Yazıcı tarafından kağıda aktarılacaktır. **Ctrl tuşu** ile birlikte kullanıldığında klavyeden girilen her veri satırı yazıcı tarafından kağıda aktarılır.



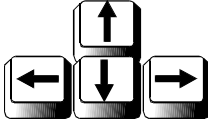
Scroll Lock tuşu, imleç ekranın en üst ya da en alt satırına geldiğinde, ekranda o an görüntülenen yazıların hareketini sağlamak için kullanılır.

Bu tuş, **Ctrl tuşu** ile birlikte kullanıldığında yürütülmekte olan program ya da komutun çalışmasını durmasını sağlar.



Pause tuşu, durdurma tuşudur. Akan görüntüyü ve çalışmakta olan programı durdurur. Tuşa tekrar basınca çalışma sürer.

Ctrl tuşu ile birlikte kullanıldığında komutun çalışmasını iptal ederek yarıda keser.



Yön Tuşları

Ok tuşlarına basıldığında imleci kendi yönünde hareket ettirir. Döküman yazımı sırasında **sol ok-sağ ok tuşu**, **Ctrl tuşu** ile birlikte kullanıldığında bir önceki-bir sonraki sözcüğe gider.



Araya Girme Tuşu

Insert tuşuna bir kez basıldığında, imleç çizgi ise yarım büyüdüğünü göreceksiniz; İmleç bütün ise yine yarım olduğunu göreceksiniz.

Bu tuşa bir kez bastıktan sonra, imlecin bulunduğu yerden itibaren, her tuş için bir karakter aralayarak araya, ekleme yapmamızı sağlar. Bir kez daha basıldığında imleç eski halini alacaktır.

Bilgisayar Burada yapılan yanlış düzeltelim. Araya **i** harfini eklememiz gerekecektir. İmleci **s** harfinin üzerine getirip **<Insert>** tuşuna bastıktan sonra **i** tuşuna basarak, **i** harfini araya sokabiliriz. Düzeltme işlemi bittikten sonra, **Ins** tuşuna bir kez daha basarak eski konuma geçmemiz gerekir



Silme Tuşu

Del ya da Delete tuşuna her basışta, imlecin üzerindeki karakter silinir. İmlecin sağındaki karakterleri kendine doğru çekerek birer karakter siler.

Bilgişayar Burada da **t** harfi fazladır. **t** harfini silelim. İmleç **t** harfinin üzerine getirilip **** tuşuna bulunduğu yerden basarsak, **t** harfi silinerek sağdaki karakterleri de kendine doğru çekecektir. Böylelikle istediğimiz düzeltmeyi yapmış olacağız.

Bazı uygulama programlarında, **Del** tuşuna basınca imlecin solundaki karakteri silerek, sağındaki karakterleri kendine doğru çeker.



Baş a Dön Tuş u

Home tuşunun işlevi, imleci ekranın sol üst köşesine götürmektir. Bazı programlarda satırın başına götürür. Döküman yazımı sırasında **Ctrl** tuşu ile birlikte kullanıldığında dökümanın en başına gider.



Sona Git Tuş u

End tuş u, imleci kullanılan programa göre ekranın ya da satırın sonuna götürür. Kullanılan uygulama programına göre değişiklik gösterir. Döküman yazımı sırasında **Ctrl** tuşu ile birlikte kullanıldığında dökümanın en sonuna gider.



Sayfa Yukarı Tuş u

Page Up tuş u, imleci bir ekran boyu kadar yukarı kaydırır. Her basışta bir önceki sayfayı görmemizi sağlar. Döküman yazımı sırasında **Ctrl** tuşu ile birlikte kullanıldığında bir önceki sayfaya gider.

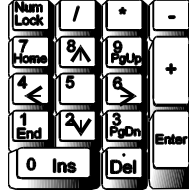


Sayfa Aşağı Tuş u

Page Down tuş u, imleci bir sayfa boyu aşağı kaydırır. Bu tuşa her basışta bir sonraki sayfayı görebiliriz. İşlevi kullanılan uygulama programına göre değişir. Döküman yazımı sırasında **Ctrl** tuşu ile birlikte kullanıldığında bir sonraki sayfaya gider.

3. Sayısal Tuşlar:

Sayısal işlemler için tasarlanmış bu bölüm, hesap makinesi şeklinde dizayn edilmiştir. Aynı zamanda edit tuşları olarak da görev yaparlar.



Num Lock tuşu, sayısal tuşların kullanıma geçmesi için kullanılır. Bu tuşun üzerinde bir ışıklı gösterge vardır. Bir kez bastığımızda bu ışıklı gösterge yanar. Bundan sonra o tuşlara yüklenen diğer görevler iptal edilir ve rakamları yazmaya geçilir. İkinci bir basışla eski konumuna gelir. Işık yanıyorsa sayısal konumda, sönmüşse edit tuşları konumundadır.

4. Fonksiyon Tuşları:

Fonksiyon tuşları programların kullanmasına kolaylık getirmek amacıyla, program tarafından değişik amaçlarla kullanılırlar:



En son girilen komutu karakter karakter tekrarlar. **F1** tuşuna her basışta daha önce belleğe yüklenen karakter dizisinden bir karakter ekrana yansıtır.



En son girilen komuttan, yeni girilen karaktere kadar kopyalama yapar. Örneğin en son komut **DISKCOPY** ise **F2 C** girildiğinde DOS ileti satırında **DISK** görüntülenir.



F3 tuşu, **F1** tuşunun yaptığı görevi bir basışta yapar. Yani belleğe yazılan bir önceki karakter dizisini bir basışta, olduğu gibi ekrana yansır.



Belirtilen karaktere kadar tüm karakterleri atlar. Örneğin en son girilen komut **TYPE DISKCOPY A: B:** ise, bu yanlış düzeltmek için, yani **DISKCOPY A: B:** yapabilmek için **F4 D F3** girişi yapılabilir.



Bir komut girişi yanlış yapılmışsa, geri geri giderek yanlış düzeltmek yerine **F5** tuşuna basarak komutu yeniden doğru olarak girebiliriz.



^Z veya kesme işaretini çıkarır.



Bir ASCII NULL karakterini satıra yerleştirir. Bu, **^@** olarak görünür ve **Ctrl @** yazımıyla aynıdır. Null karakteri bazen DOS'un satır düzenleyicisi olan EDLIN'e veri girilmesinde kullanılır.

Diğer fonksiyon tuşları kullanılan programa göre değişik görevler yüklenirler. Bu görevler programcının yüklediği görevlerdir. Kullanıcıya kolaylık olması açısından değişik fonksiyonlarda programlanabilirler.

Buraya kadar gördüğümüz tuşların kullanımında bazı aksaklıklar görürseniz bu sizi şaşırtmasın. Bu tuşların işlevleri, çeşitli Editör kullanımlarında ve program editörlerinde değişiklik gösterebilir. Özellikle paket program kullanımında bazı tuşlara değişik görevler yüklenebilir.

F. MS-DOS KOMUTLARI

Komutlar, bilgisayar ve kullanıcı arasındaki anlaşmayı sağlayarak, çalışmayı kolaylaştırırlar. İkiye ayrılırlar:

1. İç Komutlar
2. Dış Komutlar

İç ve dış komutun ne anlama geldiğini, önceki konularımızda açıklamıştık. Komutları kullanırken; komut adı, varsa sürücü, dizin adı, dosya adı, parametreler, işaret ve semboller vb. yazılarak **ENTER** tuşuna basılır. İstenen komut **A>** promptunun yanına yazılır ve **ENTER** tuşuna basılır.

5.00 ve sonraki versiyonlarla formatlanan bir disketin formatlama sırasında **UNFORMAT** ile formatlanarak, yanlışlıkla formatladığımız bir disketin içindeki bilgileri yeniden kurtarabilmemizdir. **UNDELETE** ile de bir yenilik daha eklenerek yanlışlıkla silinen bir programı yeniden canlandırabiliriz. **5.00**'la gelen bir yenilik de, komutlarda **/?** parametresini ya da **HELP Komut** şeklinde kullanarak anında yardım alınabilmesidir.

MS-DOS'un **6.20** uyarlaması her yeni uyarlamada olduğu gibi yenilikler getirmiştir. Burada **6.20** uyarlaması ile genişletilmiş yenilikleri de göreceğiz. En önemlileri belleği düzenlemesi, disk sıkıştırarak kapasitesini ortalama iki katına çıkartması, Türkçe ile ilgili yazım ve klavye desteği bulunmasıdır.

Son uyarlamalarda komutların bir çoğu menü sistemi ile çalışmaktadır. Bu da kullanıcı için büyük bir kolaylık sağlamaktadır. Menü sistemi ile çalışan bütün komutlar mouse ile de çalışabilmektedir.

Örnek:

```
A>DIR /?
ya da
A>HELP DIR    gibi...
```

MS-DOS un bir özelliği de, **DOSSHELL** kullanımı ile **DOS**'un kullanımını kolaylaştırmaktır. **DOSSHELL** ile komutları yazma yerine, pencereci menü yardımıyla çalışmasıdır. Mouse destekli de olan **DOSSHELL**, **DOS**'u komutları yazmadan kullanma olanağı sunmaktadır.

MS-DOS KOMUTLARININ YAZILIMI

MS-DOS komutlarını yazarken aşağıda vereceğimiz bilgilere uyarsak, yazım konusunda zorlanmayız.

MS-DOS işletim sisteminde komutlar, dosya iletilisinin hemen yanına yazılır. Yani **A>** ya da **C:\>** işaretinin yanına yazılır.

Bir komut yazarken önce komut, sonra sürücü adı ve : (iki nokta) işareti, sonra \ işareti ve dizin adı, (birden çok dizin adı varsa, dizin adları \ işareti ile ayrılır) daha sonra yine \ işareti ve dosya adı yazılır. Komutun anahtarını kullanacaksa dosya adından sonra / işaretinden sonra anahtarları yazarız.

Örnek: C sürücüsünde iken, komutu yazacağımızı varsayalım. Prompt kendiliğinden çıkacaktır. Yazmaya gerek yoktur. Prompt **C>** veya **C:\>** şeklinde olacaktır.

```
Komut Sürücü1: \Dizinadı\Altdizinadı\...\Dosyaadı.Uzantı
/Anahtar Sürücü2:\Dizinadı\Altdizinadı\...\Dosyaadı.Uzantı
/Anahtar
```

Yukarıdaki yazım kuralı komutun bir özelliği yoksa geçerlidir. Burada iki sürücü belirtilmiştir. İkinci kısmı kullanmazsanız, bulunduğunuz sürücüde işlem yapabilirsiniz. Komut yazımı sırasında sürücü ve dizin belirtilmezse, bulunduğunuz sürücü ve dizindeki işlemleri yapar. Bu konuda daha detaylı örnekleri komutların yazımı sırasında göreceğiz.

1. DİSKİN HAZIRLANMASI

FORMAT

Dış Komut

Görevi: Bir disket/diski bilgisayarın okuyup yazabileceği gibi düzenlemektir.

Bu komutu görmeden önce, formatlama olayı hakkında genel bir bilgi verelim:

Formatlama: Bilgisayarın yeni bir disketi kullanılır duruma getirmesidir. Yeni aldığımız boş bir disketi kullanmaya başlamadan önce formatlama'mız gerekir. Formatlama yapmadan bir disketi kullanmamız olanaksızdır. Bilgisayarın o disketi okuyabilmesi için kendi format başlığına göre izleri (Track, Sector) açması, okuyabileceği gibi uyarlaması gerekir.

Not: Eğer disketimiz dolu ise formatlandıktan sonra içindeki bilgiler silinecektir. Bunun için formatlama yapmadan önce disketi sileceğimizden emin olmalıyız. DOS 5.00 ya da daha üst bir versiyonla formatlanmışsa, herhangi bir işlem yapmadan UNFORMAT komutu ile format geri alınabilir.

FORMAT komutu, üst kısımda belirttiğimiz gibi dış komut'tur. Bunun için de iki diskete gereksinmemiz olacaktır.

1. DOS (İşletim Sistemi) Disketi.
2. Formatlanacak disket.

Yazılışı:

FORMAT [Sürücü:] [/s] [/v] [/n:xx] [/t:yy]
[/f:Kapasite] [/q] [/1] [/4] [/8]

Format komutunun anahtarları (Switch):

/s Bu anahtarı kullandığımızda, formatlama işlemi ile birlikte işletim sistemi ile ilgili iç komutların bulunduğu, bilgisayarı açmamız için gerekli dosyaları da kopyalamış oluruz. Buna **DOS-COPY** adı verilir.

Kopyalanan dosyalar şunlardır:

IO . SYS (gizli dosya)
MSDOS . SYS (gizli dosya)
COMMAND . COM

Bu dosyaları kopyalamakla bilgisayarın açılması ve sistem iç komutlarının kullanılmasını, yani bizim çalışabilmemiz için gerekli ortamı sağlamış oluruz. Gizli dosyalar **DIR** komutunu yalnız kullandığımızda görülmezler.

DOS'un 6.0 ve daha yüksek versiyonlarında formatlama yaptıktan sonra **DBLSPACE.BIN** adında bir gizli dosya daha açılır.

/v Bu anahtarı kullandığımızda formatladığımız diskete bir ad verebiliriz. /v anahtarını kullandıysak formatlama bittikten sonra karşımıza şöyle bir mesaj çıkacaktır:

Volume label (11 character, ENTER for none)?_

Bu mesaj bizden, en çok **11 karakterlik** bir ad girmemizi istemektedir. Eğer ad vermeden **ENTER**'e basarsak adsız bir disketimiz olacaktır.

Not: Kullandığımız MS-DOS'un versiyonu **4.01** ve daha yukarısı ise /V anahtarını kullanmamıza gerek kalmadan bizden diskete bir ad vermemizi isteyecektir.

/1	Disketin yalnızca bir yüzünü formatlar.
/4	Büyük kapasiteli bir disket sürücüsünde 360 Kb , çift yüzlü disketi formatlar.
/8	Bu anahtar her track'a 8 sector formatlar.
/n:xx	Belirtilen sector kadar formatlama yapar.
/t:yy	Belirtilen silindir sayısı kadar formatlama yapar.
/f:Kapasite	Belirtilen kapasitede formatlar.
/q	Bu parametre, formatın hızını artırır. Bu işlem daha önce formatlanan disket için geçerlidir.
/b	Diske MS-DOS 6.2 işletim sistemini kopyalayabilmek için, gereken boşluğu bırakarak formatlar.
/u	Diskin içindeki tüm bilgileri siler. Bu anahtarı kullandığımızda UNFORMAT komutu ile diski kurtaramayız.

a. TEK SÜRÜCÜLÜ BİLGİSAYARLARDA FORMATLAMA

Önce **DOS Sistem Disketini** sürücüye takarız.

FORMAT A: yazıp ENTER tuşuna basarız.

(A: yerine B: yazarak da tek sürücülü bilgisayarlarda formatlama yapabiliriz. Böyle yaptığımızda sanki ikinci bir sürücü varmış gibi bilgisayar bizden B disketini takmamızı isteyecektir.)

Ekranda şu mesaj belirir:

**Insert new diskette for drive A:
and press ENTER when ready**

A sürücüsüne yeni bir disket takın ve hazır olunca ENTER'e basın.

Kırmızı kullanım ışığı kapandıktan sonra, **MS-DOS** disketi sürücüden çıkartılarak, formatlanacak yeni disketi takarız. Daha sonra **ENTER'e** bastığımızda sırayla aşağıdaki ekran oluşacaktır.


```
Checking existing disk format.
Saving UNFORMAT information.

Drive A error. Cannot update the MIRROR control file.
There was an error creating the format recovery file.
This disk cannot be unformatted.
Proceed with Format (Y/N)?y
Verifying 1.44M
  xx percent completed
Format complete.
System transferred

Volume label (11 characters, ENTER for none)? SISTEM DISK ↵

  1.457.664 bytes total disk space
    198.656 bytes used by system
  1.259.008 bytes available on disk

      512 bytes in each allocation unit.
    2.459 allocation units available on disk.

Volume Serial Number is 0765-1CF5

Format another (Y/N)?
```

xx percent completed Disket formatlama sırasında çıkar. **xx** disketin % kaçının formatlandığını gösterir.

Formatlama işlemi bittiğinde ekranda şu mesaj belirir:

Format complete

Volume label (11 character, ENTER for none)?_

Buraya diskete bir isim verilmesi gerekiyorsa isim gireriz. **ENTER**'le boş geçerse isimsiz kabul edilir.

xxxxxx bytes total disk space	Disketin toplam alanı
xxxxx bytes used by system	Sistemin kapladığı alan (Sistemli formatlanmış ise)
xxxxxx bytes available on disk	Diskette kalan boşluk

xxx bytes in each allocation unit.	Bir sektöre kayıt edilebilecek byte sayısı
xxxx allocation units available on disk.	Diskette kullanılabilen alan
Volume Serial Number is xxxx-xxxx	Disketin seri numarası
Format another (Y/N)?	Başka formatlanacak disket var mı?

En son olarak da **Format another (Y/N)?** iletisi çıkacaktır. Formatlama işlemi tamamlanmıştır. Eğer formatlanacak başka bir disket var ise formatlanmış disket, sürücüden çıkartılarak formatlanacak diğer disketi takarız ve **Y** tuşuna basarak ve formatlama olayını yeniden başlatabiliriz.

Eğer başka formatlanacak disket yok ise **N** tuşuna basarak formatlama işlemi sona erdirmiş oluruz.

x.xxx bytes in bad sectors	Eğer disketimizde bozuk alan varsa, disketin alanlarını verdiğinde böyle bir ileti çıkar.
-----------------------------------	---

Formatladığımız disket daha önce sistemli formatlanmış ise, **UNFORMAT** yapılamayacağından **system transfered** mesajından önce aşağıdaki mesajla uyarılır.

**WARNING: This disk cannot be unformatted if system files are transferred.
Proceed with system transfer anyway (Y/N)?**

b. ÇİFT SÜRÜCÜLÜ BİLGİSAYARLARDA FORMATLAMA

Çift disket sürücülü bilgisayarlarda formatlama yaparken, DOS disketi A: sürücüsüne takılır. Boş olan sürücüye de formatlanacak disket takılır ve sürücü adı yerine formatlanacak disket hangi sürücüde ise o sürücünün adı yazılır.

FORMAT B: ↵

A sürücüsünde DOS disketinin bulunduğunu varsayarsak, **B** sürücüsündeki disketin formatlanmasını istemiş oluruz.

Bilgisayarımızda hard diskimiz varsa, hard diskte DOS'la ilgili bütün dosyalar yüklü ise hard diskten de formatlama yapabiliriz. Bu kez de **C:\>** promptunun yanına **FORMAT Sürücü Adı:** yazarak formatlama yapabiliriz.

c. HARD DİSKİ FORMATLAMA

Hard Disk formatlanırken aşağıdaki şekilde yazılır:

```
FORMAT C:/S ↵
```

Disket formatlamadan farklı olarak aşağıdaki mesaj çıkar:

```
WARNING: ALL DATA ON NON-REMOVABLE DISK  
DRIVE C: WILL BE LOST!  
Proceed with Format (Y/N)?_
```

Y tuşuna basarak hard diski formatlamaya başlayabiliriz. Vazgeçmek için N tuşuna basılır.

2. TARİHİ GÖSTERME - AYARLAMA

```
DATE  
İç Komut
```

Görevi: Tarihi yeniden ayarlamak ve ekranda göstermek içindir.

Yazılışı:

```
DATE [mm-dd-yy]  
[ay-gün-yıl]
```

```
DATE ↵
```

Yazıp ENTER tuşuna bastığımızda aşağıda gördüğümüz mesaj çıkar:

```
Current date is weekday mm-dd-yy Şu andaki Ay-Gün-Yıl  
Enter new date (mm-dd-yy):_ Yeni tarihi giriniz ve ↵ tuşuna  
basınız.
```

Ülke kodu olarak Türkiye seçilmişse aşağıdaki yazım şekli kullanılır:

```
DATE [dd-mm-yy]  
[gün-ay-yıl]
```

Yukarıdaki şekilde yazdığımızda bu mesajlar çıkacak, ilk satırda o anda bilgisayarın belleğinde bulunan tarih görülecektir. Biz, imlecin bulunduğu yerden sonra yeni tarihi girerek, bundan sonra tarih isteğimizde yeni tarih görüntülenecektir. Eğer yeni bir tarih girmeden ↵ tuşuna basarsak, önceki tarih yeniden bellekte kalacaktır.

DATE 7-12-1984 ↵

Bu şekilde yazılımda yazdığımız tarih belleğe yüklenerek, yeni verdiğimiz tarih bellekte yer alacaktır. Önceki yazılımdaki gibi karşımıza mesaj çıkmayacaktır.

3. ZAMANI GÖSTERME - AYARLAMA

TIME
İç Komut

Görevi: Zamanı göstermek ve ayarlamak için kullanılır.

Yazılışı:

TIME [hh:mm:ss.cc]

[Saat:Dakika:Saniye.Salise]

TIME yazıp ENTER'e bastığımızda;

Current time is hh:mm:ss.cc

Şu andaki Saat:Dakika:Saniye.Salise

Enter new time: _

Yeni zamanı giriniz ve ↵ tuşuna basınız

Mesajını alırız.

Eğer zamanı değiştirmek istiyorsak, yeni zamanı girerek ↵ tuşuna basarız. Zamanı değiştirmek istemiyorsak ↵ tuşuna basarak boş geçerse önceki zamanı kabul etmiş oluruz.

TIME 16:30:00.00 ↵

Bu yazılımda saat değiştirilerek belleğe yüklenecektir. Yine tarih değiştirmede olduğu gibi karşımıza bir mesaj çıkmayacaktır.

Aynı saati şu şekilde de girebiliriz:

TIME 16:30 ↵

4. DİSK ETİKETİ YAZMA

LABEL
Dış Komut

Görevi: Disk üzerindeki etiketi değiştirir, siler veya yaratır.

Yazılışı:
LABEL [Sürücü:] [Disk Etiketi]

Disk etiketi en çok **11 karakter** olmalıdır.

LABEL B: ↵

Yazarak **B sürücüsündeki** disketin etiketini değiştirebiliriz. Karşımıza şu mesaj çıkar:

Volume in drive B is LABEL

Volume label (11 characters, ENTER for none)?_

Etiketi değiştireceksek, bir etiket ismi yazarız (En çok 11 karakter). Eğer **ENTER**'le boş geçerse etiket ismi silineceğinden aşağıdaki mesajı verir.

Delete current volume label (Y/N)?

Y tuşuna basarsak eski etiket adı silinir, **N** tuşuna basarsak eski adı kabul etmiş oluruz.

5. MS-DOS UYARLAMA (VERSİYON) NUMARASI

VER
İç Komut

Görevi: Çalışmakta olduğumuz **MS-DOS** işletim sistemi uyarlama (Version) numarasını verir.

Yazılışı:
VER ↵

MS-DOS Version 6.20

Bu şekilde yazmakla **MS-DOS** uyarlama numarasını alabiliriz.

6. DİSK ETİKETİNİ GÖRME

VOL

İç Komut

Görevi: Varsa disk etiketini ve seri numarasını gösterir.

Yazılışı:

VOL [Sürücü:]

Belirtilen sürücüdeki disk etiketi ekranda gösterilir. Sürücü belirtilmezse çalışılan sürücüdeki diskin bilgisini verir.

VOL A: ↵

```
Volume in drive A is SISTEM DISK
Volume Serial Number is 1C47-1EDA
```

Şeklinde disk etiketini verecektir.

7. SİSTEM AKTARMA

SYS

Dış Komut

Görevi: Bir disketi **SYSTEM** disketi olarak kullanmak istediğimizde Sistem Transferi yaparız. **FORMAT** komutundaki **/S** anahtarı ile yapılan görevi yapar.

Yazılışı:

SYS [Sürücü:]

SYS A: ↵

System transfered

Yaptığımızda sistemle ilgili gizli ve açık dosyalar, **A:** sürücüsündeki diskete kopyalanacaktır. **System Transfered** mesajı ile bize transfer işleminin tamamlandığı bildirilecektir. **5.0'**dan önceki versiyonlarda yalnızca gizli dosyalar kopyalanır. **COMMAND.COM**'u ayrıca kopyalamak gerekir.

8. DOSYA ve DİZİN LİSTESİNİ GÖRME

DIR

İç Komut

Görevi: Disketin içerisinde yer alan kullanılabilir dosyaların ad ve uzantılarının tamamını, var olan özellikleriyle ekranda görüntülemeye yarar.

Yazılışı:

```
DIR [Sürücü:] [\Dizin Adı] [\Dosya Adı] [/p] [/w]
      [/o[:sıra]] [/s] [/a[:nitelik]] [/b] [/l] [/c]
```

DIR ↵

DIR komutunu yalnız başına kullanırsak, bulunduğumuz sürücünün içindeki bütün dosya isimlerini, bu dosyalara ait uzunluk, kayıt tarihi ve zamanı ile ilgili bilgileri ekranda listeler halinde verir.

Anahtarlar:

/p Dosya listesini, sayfalar halinde (ekran boyu) verir. Disketimizde çok sayıda dosya kayıtlı ise **DIR** komutunu yalnız başına kullandığımızda hızlı bir akışla geçeceğinden, dosyaları kontrol etmek ve okumak olanaksızdır. Böyle durumlarda **/p** anahtarını kullanarak, sayfa sayfa akışı sağlamış oluruz ve rahatlıkla okuyabiliriz. Her sayfa sonunda bir sonraki sayfaya geçmek için herhangi bir tuşa basmamız gerekir.

Sayfalar arasında **Press any key to continue...** mesajı, devam için herhangi bir tuşa basın anlamındadır.

/w Dosya adlarını sıkışık olarak listeler. Bu anahtarı kullandığımızda yalnızca dosya adlarını vererek, aynı anda daha çok dosya adını görmemizi sağlar. Dosya adları ekranda beş sütun halinde görülür.

/s Tüm alt dizinleri de arattırır.

/o Dosyaların isim, tarih ve uzunluk ölçülerine göre sıralanmasını sağlar.

- /a** Gizli dosyaların da listelenmesini sağlar.
:nitelik
H Gizli dosyalar.
-H Gizli olmayan dosyalar.
S Sistem dosyaları.
-S Sistem dosyası olmayanlar.
D Dizinler.
-D Yalnızca dosyalar. Bu anahtarla dizinler listelenmezler.
A Arşiv dosyaları.
-A Arşiv dosyası olmayanlar.
R Yalnızca okunabilir. dosyalar.
-R Yalnızca okunabilir olmayan dosyalar.
- /b** Yalnızca dosya adlarının görüntülenmesini sağlar.
- /l** Dosya adlarının küçük harfle görülmesini sağlar.
- /c** Disk **Dblspace** kullanılarak sıkıştırılmış ise, dosyaların sıkıştırma oranlarını gösterir.

Bu anahtarları birlikte de kullanabiliriz.

DIR ↵

ya da

DIR *.* ↵

Dosya adı ve uzantısı ne olursa olsun, bütün dosyaların listesini verir.

Ekran çıktısında sütun olarak sırasıyla dosya ve dizin adları, kapladıkları byte olarak alan, kayıt tarihleri ve kayıt saatleri verilmiştir. Uzantısı **<DIR>** şeklinde olanlar dizinlerdir. Diğerleri dosyadır.

Dosya ve dizin adlarının en altında dosya ve dizin sayısı ve karşısında o dizinde bulunan dosyaların kapladığı alan byte olarak çıkar. En altında ise diskte kalan boşluk yine byte olarak verilir.

Görüntü şu biçimde olur:

```
Volume in drive A is SISTEM DISK
Volume Serial Number is 1C47-1EDA
Directory of A:\
AUTOEXEC  BAT           494 07/07/94   23:18
CHKDSK    EXE          12.241 30/09/93    6:20
COMMAND   COM           54.619 30/09/93    6:20
CONFIG    SYS            284 01/06/94   20:24
DILLER    <DIR>           24/06/94   21:16
DISKCOPY  COM          13.335 30/09/93    6:20
DOS       <DIR>           19/06/94   23:17
FKLAVYE   COM           2.486 01/01/80    0:02
FORMAT    COM          22.916 30/09/93    6:20
LABEL     EXE           9.390 30/09/93    6:20
MEKTUP    <DIR>          20.713 13/12/93   22:23
OYUN      <DIR>           19/06/94   23:19
PW        <DIR>           20/06/94   21:32
WINA20    386           9.349 11/08/93    6:20
WINDOWS   <DIR>          19/06/94   23:19
XCOPY     EXE          16.930 30/09/93    6:20
16 file(s)
162.757 bytes free
367.104 bytes free
```

Disk Etiketi

Dosya Adları Dosya ve Dizin Sayısı Dosya Uzunlukları Kayıt Tarihleri Diskte Kalan Boş Yer Dosyaların Kapladığı Alan Kayıt Saatleri

DIR MUHASEBE.* ↵

Dosya adı **MUHASEBE** olan, uzantısı ne olursa olsun bütün dosyaların adlarını listeler.

DIR B*.* ↵

DIR BOR*.* ↵

Birinci örnek dosya adı **B** ile başlayan, ikinci örnek dosya adı **BOR** ile başlayan ve uzantısı ne olursa olsun bütün dosyaları listeler.

DIR *.EXE ↵

Dosya adına bakılmaksızın, uzantısı **EXE** olan bütün dosyaların listesini verir.

DIR M???.* ↵

M harfi ile başlayan ve **dört harfli** olan dosyaların listesini verir.

DIR>PRN ↵ ya da DIR>LPT1: ↵

Çalıştığımız sürücüdeki dosyaların listesini yazıcıdan verir.

Yukarıda verdiğimiz örnekler çalıştığımız sürücü ve dizin için geçerlidir. Çalıştığımız sürücüden, diğer bir sürücü ve dizindeki dosyaları görmek için, görmek istediğimiz sürücünün ve dizinin adını eklemeliyiz.

DIR B: ↵

DIR C:\BASIC ↵ gibi...

DIR A:\PW ↵ yaptığımızda,

```
Volume in drive A is SISTEM DISK
Volume Serial Number is 1C47-1EDA
Directory of A:\PW

.                <DIR>                20/06/94    21:32
..               <DIR>                20/06/94    21:32
DATA             <DIR>                20/06/94    21:33
KITAP            <DIR>                20/06/94    21:36
OZEL             <DIR>                20/06/94    21:33
PFS              DIR                236 01/01/80    0:28
PW               PRG                318.505 04/02/88    0:16
PW               HLP                22.409 04/02/88    0:16
PW               PRI                11.396 04/02/88    0:17
PW               COM                4.051 04/02/88    0:09
PW               SET                1.470 09/04/94   12:12
PW               PIF                369 04/02/88    0:17
PW               LML                52.080 02/01/80   15:37
PW               MAC                1.834 11/03/93   12:36
PW               NET                60 20/04/93     13:10
15 file(s)      412.410 bytes
367.104 bytes free
```

Şeklinde görülür.

. <DIR> Dosya ve dizin adlarının en üstünde yukarıdaki işaretler
.. <DIR> varsa, bunlar bizim bir dizin içerisinde olduğumuzu gösterir.

Bu örneklerin sonuna isteğe göre /P, /S, /B, /A, /L, /C, /O ve /W parametrelerini de ekleyebilirsiniz.

DIR /W ↵ yaptığımızda görüntü aşağıdaki gibi olacaktır.

```
Volume in drive A is SISTEM DISK
Volume Serial Number is 1C47-1EDA
Directory of A:\

AUTOEXEC.BAT      CHKDSK.EXE      COMMAND.COM      CONFIG.SYS      [DILLER]
DISKCOPY.COM      [DOS]          FKLAVYE.COM      FORMAT.COM      LABEL.EXE
MEKTUP           [OYUN]         [PW]             WINA20.386     [WINDOWS]
XCOPY.EXE
      16 file(s)          162.756 bytes
                          367.104 bytes free
```

Burada da [] işaretleri arasında olanlar dizinlerdir.

(Yaptığımız bütün işlemleri belleğe göndermemiz ve sonuç alabilmemiz için ENTER tuşuna mutlaka basmalıyız. Bu işlem bundan sonraki bütün komutlar için geçerlidir.)

TREE

Dış Komut

Görevi: Sürücüdeki disk/disketin dizin ve alt dizinlerini bir şema halinde ekranda gösterir. İsteğe göre içerik listesi verir.

Yazılışı:

TREE [Sürücü:] [Dizin Adı] [/f] [/a]

Anahtarlar:

/f Her dizindeki dosyaların isimlerini verir.

/a Hızlı bir yazılım sağlayarak, kod sayfalarında bulunan grafik karakterlerin kullanılmasını sağlar.

TREE A: ↵ yaptığımızda aşağıdaki gibi ağaç şeklinde dizin listesi gelir.

```
Directory PATH listing for Volume SYSTEM DISK
Volume Serial Number is 1C47-1EDA
A:
├── DOS
├── PW
│   ├── DATA
│   ├── OZEL
│   │   ├── İSYAZI
│   │   │   ├── KAMU
│   │   │   └── SİRKET
│   │   └── KİSİSEL
│   └── KİTAP
├── OYUN
│   ├── PRENS
│   └── TETRİS
├── DİLLER
│   ├── BASIC
│   ├── COBOL
│   └── PASCAL
├── WINDOWS
│   ├── SYSTEM
│   ├── CORELDRW
│   └── WINWORD
```

Bulduğumuz sürücünün dışındaki bir sürücüde ve bir dizinin alt dizinlerini de görmek isteyebilir. Burada A sürücüsünde bulunan **PW** dizininin alt dizinlerini görmek isteyelim:

TREE A:\PW ↵ yaptığımızda,

```
Directory PATH listing for Volume SYSTEM DISK
Volume Serial Number is 1C47-1EDA
A:\PW
├── DATA
├── OZEL
│   ├── İSYAZI
│   │   ├── KAMU
│   │   └── SİRKET
│   └── KİSİSEL
└── KİTAP
```

Şeklinde görüntü çıkar.

9. EK RANI SİLME

CLS

İç Komut

Görevi: Ekranı temizler.

Yazılışı:

CLS ↵

Bu komut ile ekrandaki bütün yazılar silinir. Bu arada bir konuya dikkatimizi çekelim. Ekranın temizlenmesi ile bellekteki bilgiler kaybolmaz. Yalnız ekrandaki yazılar silinir ve imleç (kursör) başa gelir.

10. DİZİN AÇMA, DEĞİŞTİRME ve SİLME

a. DİZİN AÇMA

MD

MKDIR

İç Komut

Görevi: Dizin (Directory) yaratır.

Yazılışı:

MKDIR [Sürücü:] \Dizin Adı

veya

MD [Sürücü:] \Dizin Adı

Dizinlere istenirse 3 karakterlik uzantı verilebilir.

MD \MUHASEBE ↵

veya

MD MUHASEBE ↵

MUHASEBE adında bir dizin açar. **MD**'den sonra \ işareti yerine boşluk koyarak da yazabiliriz.

MD \MUHASEBE \AFIRMA ↵

MUHASEBE dizini altında **AFIRMA** adında bir alt dizin yaratır.

Bulduğumuz sürücüye değil de başka bir sürücüde dizin açmak istersek aşağıdaki biçimde yazarız:

MD C:SEKRETER ↵

C sürücüsüne **SEKRETER** adında bir dizin açar.

Daha önce **A sürücüsünde PW** adında bir dizin açmış olduğumuzu varsayarak, **PW**'nin altına **YAZISMA** adında bir dizin açmak istersek aşağıdaki biçimde yazarız:

MD A:\PW\YAZISMA ↵

b. DİZİN DEĞİŞTİRME

CD CHDIR Komut	İç
---	-----------

Görevi: Bir dizinden, diğer bir dizine geçmeye yarar.

Yazılışı:

CHDIR [Sürücü:] [\Dizin Adı]

ya da

CD [Sürücü:] [\Dizin Adı]

CD ↵

Çalıştığımız dizinin adını gösterir.

CD\MUHASEBE ↵

Bulduğumuz dizinden **MUHASEBE** adlı dizine geçer.

CD.. ↵

Bulduğumuz dizinden çıkarak bir üst dizine geçer. Yani bir alt dizinden bir üst dizine geçer. Çalıştığımız dizin, dizin ise ana dizine, alt dizin ise, alt dizinin bulunduğu dizine geçer.

CD\MUHASEBE ↵

Bulduğumuz dizinden **MUHASEBE** adlı dizine geçer.

CD\MUHASEBE\AFIRMA ↵

Bulduğumuz dizinden, **MUHASEBE** dizininin alt dizini olan **AFIRMA** dizinine geçer.

```
CD\ ↵
```

Bulduğumuz dizinin adını verir. Eğer alt dizinde isek, bütün dizinlerden çıkarak ana dizine geçer.

Bir dizin içerisinde iken, başka bir dizine geçmek için mutlaka **CD**'den sonra \ işaretini yazmamız gerekir.

Örnek verecek olursak, **MUHASEBE** dizini içinde iken **SEKRETER** dizinine geçmek istediğimizde aşağıdaki yazım biçimini kullanmamız gerekir.

```
CD\SEKRETER ↵
```

c. DİZİN ADI SİLME

RD RMDIR Komut	İç
---	-----------

Görevi: Bir dizinin adını disketten siler.

Yazılışı:

```
RMDIR [Sürücü:] <Dizin Adı>
```

ya da

```
RD [Sürücü:] <Dizin Adı>
```

```
RD MUHASEBE ↵
```

MUHASEBE adlı dizinin adını, disketten siler.

```
RD C:\SEKRETER\SINIF1 ↵
```

C sürücüsündeki SEKRETER dizini altındaki **SINIF1** dizinini siler.

Bir dizini silmeden önce, dizin içindeki dosyaları silmemiz gerekir.

Eğer dizin içindeki dosyaları da silmek istiyorsak, dizin içine girerek **DEL** komutuyla silmemiz gerekir.

Örnek:

CD \MUHASEBE ↵ **MUHASEBE** dizinine geçilir.
DEL *.* ↵ **MUHASEBE** dizinindeki bütün dosyalar silinir.
CD . . ↵ Bulduğumuz dizinden çıkılır.
RD MUHASEBE ↵ **MUHASEBE** dizini silinir

Bir dizinin içindeki dosyaları dışarıdan şu şekilde de silebiliriz:

DEL MUHASEBE ↵

Yazdığımızda **MUHASEBE** dizini içindeki dosyaların hepsini siler. Bunu yaparken **DEL** komutunu kullandığımız için, silinip silinmeyeceği bize sorulur. Sonra **RD** ile Dizin Adını yazarak, dizini tamamen sileriz.

İleride göreceğimiz **DELTREE** komutu belirtilen dizinin altında bulunan bütün dizin ve dosyaları tek bir komutla siler.

11. DOSYA KOPYALAMA

COPY
komut

İç

Görevi: Bir veya daha çok dosyayı başka bir diskete kopyalamaya yarar. Aynı disket üzerinde de kopyalama yapılabilir.

Yazılışı:

COPY [Sürücü:] [\Dizin Adı] [\Dosya Adı] [Sürücü:]
[\Dizin Adı] [\Dosya Adı] [/a] [/b] [/v]

Anahtarlar:

/a **ASCII** dosyalarını kopyalamaya izin verir. Dosya ASCII değil ve herhangi bir yerde dosya sonu işareti varsa, dosyayı eksik kopyalar.

Kaynak dosyanın önüne yazılırsa: Dosyayı ASCII dosya gibi görmesini sağlar. İlk dosya sonu işareti hariç, dosya kopyalanır.

Hedef dosyanın önüne yazılırsa: Dosya sonu işaretini dosyanın en son karakteri gibi eklenmesini sağlar.

/b Dosyada, dosya sonu işareti olsa bile, dosyanın tamamını kopyalar.

Kaynak dosyanın önüne yazılırsa: Dosya sonu işareti dahil dosyanın tamamının kopyalanmasını sağlar.

Hedef dosyanın önüne yazılırsa: Bir dosya sonu karakterini eklemes.

- /v Kopyalamayı kontrollu yapar. Dosyaların doğru kopyalandığından emin olur.
- /y Kopyalama sırasında hedeflenen ortamda aynı isimde dosya varsa üzerine kopyalanıp kopyalanmayacağını sormaması için bu anahtarı kullanırız.
- /-y /Y anahtarının tersini yapar, üzerine yazılıp yazılmayacağını sorar.

COPY UCRET.BAS UCRET2.BAS ↵

UCRET.BAS isimli dosyayı, bulunduğu ortama **UCRET2.BAS** isimli yeni bir dosya açarak kopyalar. Dosyaların içerikleri aynı olacağından yalnızca adı değişik olacaktır. Yani aynı içerikli iki adet dosya elde etmiş olacağız.

COPY MUHASEBE.EXE B: ↵

MUHASEBE.EXE isimli dosyayı **B sürücüsüne** kopyalar. Yanlışlıkla sürücü adı yerine çalıştığımız sürücünün adını verirsek kopya olayı gerçekleşmeyeceğinden hata mesajı verecektir.

COPY UCRET.BAS B:YENIUCR.BAS ↵

UCRET.BAS isimli dosyayı **B sürücüsüne YENIUCR.BAS** adı altında kopyalar.

COPY komutunda da * ve ? jokerlerini işimize ve isteğe bağlı olarak kullanabiliriz.

COPY *.* C: ↵

Bulduğumuz ortamdaki bütün dosyaları **C sürücüsüne** kopyalar.

COPY *.EXE B: ↵

Bulduğumuz ortamda, uzantısı **EXE** olan bütün dosyaları **B sürücüsüne** kopyalar.

COPY MUH*.* B: ↵

Bulduğumuz ortamdaki **MUH** ile başlayan, uzantısı ne olursa olsun bütün dosyaları **B sürücüsüne** kopyalar.

COPY UCR*.INT C: ↵

Bulduğumuz ortamda ilk üç harfi **UCR** ve uzantısı **INT** olan bütün dosyaları **C sürücüsüne**, yani **Hard Disk'e** kopyalar.

Copy komutu ile birlikte + işareti de kullanılır. Örnekle gösterecek olursak:

COPY BORDRO1.EXE+BORDRO2.EXE ↵

Biçiminde yazdığımızda **BORDRO2.EXE** dosyası, **BORDRO1.EXE** dosyasının sonuna eklenir.

COPY BORDRO1.EXE+BORDRO2.EXE BORDRO.EXE ↵

Biçiminde yazılıştta ise **BORDRO1.EXE** ve **BORDRO2.EXE** programları birleştirilerek **BORDRO.EXE** adı ile yeni bir dosya oluşturularak kopyalanır.

Eğer başka bir sürücüde bulunan bir dizine kopyalama yapacaksak:

COPY BORDRO.EXE C:\ISCI ↵

Biçiminde yazarsak, **BORDRO.EXE** adlı programı **C sürücüsünde** bulunan **ISCI** adlı dizine kopyalayacaktır.

COPY *.* C:\ISLETME ↵

Biçiminde yazdığımızda bulunduğumuz sürücüdeki bütün dosyaları **C sürücüsündeki ISLETME** adındaki dizine kopyalayacaktır.

COPY C:\SEKRETER*.* ↵

Bu şekilde yazdığımızda, **C sürücüsündeki SEKRETER** dizininde bulunan bütün dosyalar bulunduğumuz ortama kopyalanacaktır.

COPY A:\PW\DATA*.DAT C:\YAZI ↵

A sürücüsündeki PW dizinin altındaki **DATA** alt dizinindeki **DAT** uzantılı dosyaları **C sürücüsündeki YAZI** dizinine kopyalar.

Hedef sürücü ve dizinde aynı ad ve uzantıda dosya varsa, üzerine yazılıp yazılmayacağı aşağıdaki mesaj ile uyarılır.

Overwrite X:\Dizinadı\Dosyaadı (Yes/No/All) ?

Yalnız adı görülen dosyanın üzerine yazılacaksa **Y** tuşuna, yazılmayacaksa **N** tuşuna, bütün dosyaların üzerine yazılacaksa **A** tuşuna basılır. **A** tuşuna basarsak bir daha soru sormaz. Diğer tuşlara basarsak aynı ada ve uzantıdaki bütün dosyalar için soracaktır.

XCOPY

Dış Komut

Görevi: Disk/diskette bulunan dizin ve alt dizinler dahil olmak üzere dizinleri ve dosyaları kopyalar.

Yazılışı:

**XCOPY [Sürücü:] [\Dizin Adı\Dosya Adı] [Sürücü:]
[\Dizin Adı\Dosya Adı] [/a] [/d:Tarih]
[/e] [/m] [/p] [/s] [/v] [/w]**

XCOPY A:*. * C:\YENIMUH ↵

A sürücüsündeki diskette bulunan tüm dosyaları, **C sürücüsünde** yani hard diskte bulunan **YENIMUH** adlı dizine kopyalar.

XCOPY komutunun COPY komutuna göre avantajlı yanı, daha hızlı kopyalamasıdır. Önce dosyaları belleğe okur, sonra kayıt eder. Okuma sırasında aşağıdaki mesaj çıkar:

Reading source file(s)...

Bu mesajla dosyalar okunduktan sonra, hedef ortama kopyalama yapılacaktır.

Anahtarlar:

/a Arşiv niteliğindeki dosyaları kopyalar.

/d:Tarih Belirli bir tarihten sonra değiştirilmiş dosyaları kopyalar.

/e	Bu anahtar /s anahtarı ile birlikte kullanılmalıdır. Boş olan dizinleri de kopyalar.
/m	Arşiv niteliğindeki dosyaları kayıt ettikten sonra, dosyanın arşiv niteliğini iptal eder.
/s	Bu anahtar kullanılmadığı takdirde komut tek bir dizin için çalışır. Bu komutu kullanarak alt dizinleri de kopyalayabiliriz.
/v	Her dosyayı, hedef dosyada yazılı olduğu biçimde onaylar. Hedef dosyanın kaynak dosya tarafından kabul edilmiş olduğunu bildirir.
/w	Kopyalama işlemine başlamadan önce beklemesini sağlar.
/p	Bir dosyayı kopyalamadan önce uyarır.
/y	Kopyalama sırasında hedeflenen ortamda aynı isimde dosya varsa üzerine kopyalanıp kopyalanmayacağını sormaması için bu anahtarı kullanırız.
/-y	/Y anahtarının tersini yapar, üzerine yazılıp yazılmayacağını sorar.

XCOPY komutunu kullanırken karşı tarafta istediğiniz isimde dizin yoksa, XCOPY bunu da çözümler. Aşağıdaki şekilde yazımla bunu gerçekleştirebiliriz.

```
XCOPY *.* A:\SEKRETER ↵
```

Kopyalama başlayınca aşağıda verilen mesaj çıkar:

```
Does SEKRETER specify a file name  
or directory name on the target  
(F = file, D = directory)?
```

Burada kopyalanacak dosyaların dosya olarak mı, yoksa dizin olarak mı kayıt edileceği sorulmaktadır. **D** tuşuna basarsanız, hedef diskette **SEKRETER** adında bir dizin açarak, dosyaları açtığı dizinin içinesine kopyalayacaktır.

Hedef sürücü ve dizinde aynı ad ve uzantıda dosya varsa, üzerine yazılıp yazılmayacağı aşağıdaki mesaj ile uyarılır.

```
Overwrite X:\Dizinadı\Dosyaadı (Yes/No/All)?
```

Yalnız adı görülen dosyanın üzerine yazılacaksa **Y** tuşuna, yazılmayacaksa **N** tuşuna, bütün dosyaların üzerine yazılacaksa **A** tuşuna basılır. **A** tuşuna basarsak bir daha soru sormaz. Diğer tuşlara basarsak aynı ada ve uzantıdaki bütün dosyalar için soracaktır.

12. DOSYA TAŞIMA ve DİZİN ADI DEĞİŞTİRME

MOVE

Dış Komut

Görevi: Dosya ya da dosyaları bir yerden başka bir yere taşır ve dizin adı değiştirir. Eski dosyalar taşınacağından, yeni yerinde olacak ve eski yerinde bulunmayacaktır.

Yazılışı:

```
MOVE [/y/-y] [Sürücü:] [\DizinAdı\DosyaAdı]  
[ , [Sürücü:] [\DizinAdı] ... [\Hedef]
```

Anahtarlar:

- /y Taşıma sırasında hedef dizinde aynı adlı dosyalar varsa, taşınıp taşınmayacağını sorar. Bu soruyu önlemek için /y anahtarı kullanılır.
- /-y Taşıma sırasında aynı hedef dizinde aynı adla dosya varsa, üzerine yazılıp yazılmayacağını sorar. /y anahtarının terisini yapar.
- Hedef** Buraya yazılacak bir isimle, taşıma yerine yeni bir dizin açılarak oraya kayıt yapılır.

```
MOVE C:\PW\*.* C:\PWYENI
```

C sürücüsündeki PW dizininde bulunan dosyalar, yine C sürücüsündeki PWYENI dizinine taşınacaktır.

```
c:\pw\pw.prg => c:\pwyeni\pw.prg [ok]  
c:\pw\pw.hlp => c:\pwyeni\pw.hlp [ok]  
c:\pw\pw.net => c:\pwyeni\pw.net [ok]  
Overwrite c:\pw\pfs.dir (Yes/No/All)?a  
c:\pw\pfs.dir => c:\pwyeni\pfs.dir [ok]  
c:\pw\chklist.ms => c:\pwyeni\chklist.ms [ok]
```

Şeklinde çıkarak taşıma işlemi yapılacaktır.

Dizin adı değiştirmek için ise aşağıdaki şekilde yazmak gerekir:

```
MOVE PWYENI PW2
```

Şeklinde yazdığımızda aşağıdaki mesaj ile değişiklik onaylanacaktır.

```
c:\pwyeni => c:\pw2 [ok]
```

13. AYNI İSİMDEKİ DOSYALARI KOPYALAMA

REPLACE

Dış Komut

Görevi: Aynı isimdeki dosyaları hedef sürücüye kopyalar.

Yazılışı:

```
REPLACE [Sürücü1:]\Dizinadı\Dosyaadı\  
[Sürücü2:]\Dizinadı\Dosyaadı [/a] [/p] [/r] [/s] [/w] [/u]
```

Anahtarları:

- /a Hedef dizindeki dosyaları değiştirmek yerine ekleme yapılmasını sağlar. /s ve /u anahtarlarıyla birlikte kullanılmaz.
- /p Dosyaları ekleme ya da değiştirme yapmadan önce mesajla uyarır.
- /r Bu anahtarı kullanırsanız silinemez dosyaları da değiştirir.
- /s Hedef dosyanın alt dizinlerini de araştırır. /a anahtarı ile birlikte kullanılmaz.
- /w Dosya değiştirme işlemi başlamadan önce uyarır.
- /u Yalnızca kaynak dizindeki dosyalardan, eski tarihli dosyaları değiştirir. /s ve /a ile birlikte kullanılmaz.

14. DİSKET KOPYALAMA

DISKCOPY

Dış Komut

Görevi: Bir kaynak diskette bulunan dosyaların hepsini, hedef diskete kopyalamak için kullanılır. **DISKCOPY** yapılacak disketlerin boyutları ve kapasiteleri aynı olmalıdır.

Yazılışı:

```
DISKCOPY [Sürücü1:] [Sürücü2:] [/1] [/v] [/m]
```

Anahtarlar:

- /1 Disketin yalnızca bir yüzünü kopyalar.
- /v Kopyalamanın doğru yapıldığını kontrol eder.
- /m DOS 6.2 disket kopyalamayı bir defada yapmaktadır. Eski yöntemle kopya yapılmasını istiyorsak bu anahtarı kullanırız.

DISKCOPY A: A: ↵

Bilgisayarımız tek sürücülü ise, sürücüye sistem disketini takarak bu şekilde yazıp **ENTER** tuşuna basarız. Bilgisayar **DISKCOPY** bilgilerini belleğine yükledikten sonra şu mesajı verir:

```
Insert SOURCE diskette in drive A:
Press any key to continue . . .
```

Bunun anlamı **Kaynak Disketi**, yani kopyası alınacak disketi **A sürücüsüne** takın ve herhangi bir tuşa basın. Kaynak disket, kopyası alınacak diskettir.

Biz, kopyasını almak istediğimiz disketi, **A sürücüsüne** takarız ve herhangi bir tuşa basarız. Kopyalama işlemi sırasında şu mesaj çıkar:

```
Copying 80 tracks, 18 sectors per track, 2 side(s)
Reading from source diskette . . .
```

Bilgisayar, kaynak disketteki bilgileri belleğine aldıktan sonra, yeni bir mesaj verecektir.

```
Insert TARGET diskette in drive A:
Press any key to continue . . .
```

Bizden bu kez, **Hedef Disket'i**, yani kopyanın yapılacağı disketi istemektedir. Hedef disket, boş olan yani üzerine kopya yapılacak diskettir.

Bu kez kaynak disketi çıkarıp, yerine hedef disketi takarız ve herhangi bir tuşa basarız. Hedef disketimize yazılmanın yapıldığını aşağıdaki mesajla bildirir.

```
Writing to target diskette . . .
```

Hedef disketimiz formatlı değilse, formatlanarak kopyalama yapılacağından aşağıdaki mesaj çıkar.

```
Formatting while copying
```

Bu işlemin birkaç kez yinelenmesi istenebilir. Bu da bilgisayarımızın kapasitesine bağlıdır. Belleği büyükse, daha kısa zamanda kopyalama işlemi tamamlanacaktır. Kopyalama işlemi tamamlandıktan sonra bize şu mesajı verecektir:

Disketin kopyalanması bittikten sonra aşağıdaki mesaj çıkacaktır.

```
Do you wish to write another duplicate of this disk (Y/N)?_
```

Aynı kaynak disketi bir başka diskete daha kopyalayacaksa bu mesaja **Y** ile, kopyalama yapmayacaksa **N** ile yanıt veririz.

```
Volume Serial Number is 1ECE-424E
Copy another diskette (Y/N)? n
```

Bu mesaj kopyalama işleminin bittiğini bildirir. Eğer başka kopya alacaksa, **Y** tuşuna basarak kopya alma işlemini sürdürebiliriz. Başka kopya almayacaksa, **N** tuşuna basarak işlemi bitiririz.

```
DISKCOPY A: B: ↵
```

Çift disket sürücülü bilgisayarlarda kopyalama işlemi için bu yazılımı kullanırız. **A sürücüsüne** sistem disketini takarız. **DISKCOPY** yüklenir ve sonra kaynak disketi; **A sürücüsüne**, hedef disketi **B sürücüsüne** takarak kopyalamayı yapabiliriz. Çift sürücülü bilgisayarlarda kopyalama işlemi daha kolay olacaktır. Az önceki örneğimizde olduğu gibi disket takıp-çıkarma olayı burada söz konusu değildir. Bilgisayar kendiliğinden **A sürücüsünden** okuyacak, **B sürücüsüne** kopyalayacaktır.

Hedef disketimiz formatlı değilse **DISKCOPY** komutu, hedef disketi önce formatlayacak sonra kopyalama işlemini yapacaktır. Ama bunu kullanmamanızı öneririz. Eğer disketinizde kötü sector varsa, bu şekilde bir kopyalama başarısız olur. **FORMAT** komutu bunları kontrol ettiğinden, kötü sektörlerden bilginiz olacaktır.

15. DOSYA SİLME

DEL ERASE Komut	İç
--	-----------

Görevi: Disketten dosya silmeye yarar.

Yazılışı:

```
DEL [Sürücü:] [\Dizin Adı] [\Dosya Adı] [/p]
```

/p Bu parametreyi kullanarak dosyaları silerken bize sormasını sağlarız. Birçok dosyayı silmesini istediğimizde, bazılarını silmesini istemiyorsak bu parametreyi kullandığımızda, **N** yanıtı ile o dosyanın silinmesini engellemiş oluruz.

```
DEL BORDRO.EXE ↵
```

Bu yazılımla **BORDRO.EXE** adlı dosya disketten silinir.

Bu komutu kullanırken * ve ? jokerlerini daha önceki örneklerimizde olduğu gibi işimizin uygunluğuna ve isteğe göre kullanabiliriz.

DEL *.* ↵

Disketteki bütün dosyaları silmek için bu şekilde yazarız. Tüm dosyaları silmek istediğimizde karşımıza şu mesaj çıkar:

**All files in directory will be deleted!
Are you sure? (Y/N)**

Dizindeki tüm dosyaları sileceğinizden, Emin misiniz? (Y/N)

Bu mesaj bizim için bir uyarıdır. Olası bir yanlışlığı düşünerek, bizden silme olayını yeniden doğrulamamızı ister. Yanıt olarak **Y** girersek silme olayı gerçekleşecek, **N** girersek silme işleminden vazgeçilecektir.

DEL *.BAS ↵

BAS uzantılı dosyaları disketten siler.

DEL B:MEKTUP.BAK ↵

Yazılışıyla **B** sürücüsündeki **MEKTUP.BAK** adlı dosyayı siler.

DEL A:*. * ↵

Şeklinde yazarak **A** sürücüsündeki bütün dosyaları silebiliriz.

DEL C:\SEKRETER*.BAK

Şeklinde yazarak **C** sürücüsündeki **SEKRETER** dizininde bulunan uzantısı **.BAK** olan tüm dosyaları siler.

16. DİZİN SİLME

DELTREE

Dış Komut

Görevi: Dizini tüm dosyaları ve alt dizinleri ile birlikte siler.

Yazılış:

DELTREE /y Sürücü:Dizinadı

Anahtar:

/y Silme sırasında onay almaz.

Örnek:

DELTREE A:PW ↵

A sürücüsünde bulunan **PW** dizinini sizden onay bekleyerek tüm dosya ve alt dizinleri ile birlikte siler.

17. DOSYA İÇERİĞİNİ GÖRME

TYPE

İç Komut

Görevi: Dosyanın içeriğini ekranda göstermeye yarar. Bu komutu kullanırken unutmayacağımız nokta, görmek istediğimiz dosyanın içeriğinin bizim anlayabilmemiz için bir Editör'le yazılmış olması, en önemlisi **ASCII** kodda kaydedilmiş olması gerekir. Derlenmiş ya da herhangi bir (BASIC gibi...) yorumlayıcı (Interpreter) ile yazılmış dosya içerikleri bizim anlayabileceğimiz dille değildir. Böyle dosyaların içeriğini **TYPE** komutuyla görmek istediğimizde karakter kodlarıyla karşılaşırız. **BASIC**'te yazılmış bir programı **TYPE** komutuyla işletim sisteminde görmek istiyorsak **ASCII** kodda kaydetmeliyiz.

Yazılışı:

TYPE [Sürücü:] [\Dizinadı\] [Dosya Adı]

TYPE SIIR.BAK ↵

SIIR.BAK adlı dosyanın içeriğini ekranda görüntüler.

Bir dosyanın içeriğini ekranda bu şekilde yazarak görmek istediğimizde, dosyanın içeriği uzunsa gözümüzün önünden hızla akacaktır. Bunu önlemek için sonuna **MORE** komutunu eklersek herhangi bir tuşa basıncaya kadar dosya içeriği sayfa boyu ekranda görüntülenir. Herhangi bir tuşa basınca bir sonraki sayfayı görüntüler. Görme sırasında herhangi bir yerde durdurmak istersek **PAUSE** tuşuna ya da **Ctrl+NumLock** tuşlarına basarak kayan görüntüyü herhangi bir tuşa basana kadar durdurabiliriz.

TYPE SIIR.BAK|MORE ↵

Şeklinde yazarak görüntüyü sayfa boyu durdurabiliriz. **MORE** komutu dış komuttur. Bu nedenle **DOS** disketinden çalıştığımız diskete kopyalamamız gerekir.

TYPE>PRN SIIR.BAK ↵ ya da **TYPE SIIR.BAK >LPT1 ↵**

SIIR.BAK adlı dosyanın içeriğini yazıcıdan alabiliriz.

18. DOSYA ADI DEĞİŞTİRME

REN
RENAME
İç Komut

Görevi: Diskette kayıtlı bulunan dosyaların adlarını değiştirmek için kullanılır.

Yazılışı:

```
REN [Sürücü:] [\Dizin Adı] [\Eski Dosya Adı]
                                     [Yeni Dosya Adı]
RENAME [Sürücü:] [\Dizin Adı] [\Eski Dosya Adı]
                                     [YeniDosya Adı]
```

Bu yazılımların ikisi de aynı görevi yapar.

```
REN MUH.BAS MUHASEBE.BAS ↵
```

MUH.BAS isimli dosyayı MUHASEBE.BAS olarak değiştirir.

Bu komutla birlikte * ve ? jokerlerini isteğe göre kullanabiliriz.

```
REN *.ESK *.YEN ↵
```

Disketteki **ESK** uzantılı bütün dosyaların uzantılarını **YEN** olarak değiştirir.

19. DİSK KONTROLÜ

CHKDSK
Dış Komut

Görevi: Disk/Disketi kontrol eder. Okunamaz alanları belirler ve kullanılır duruma getirir.

Yazılışı:

```
CHKDSK [Sürücü:] [\Dizin Adı] [\Dosya Adı] [/f] [/v]
```

Anahtarlar:

- /f Disketteki hatalı alanları belirler ve temizleyerek kullanılabilir duruma getirir.
- /v Disketteki hatalı dizin ve dosyaları kontrol eder ve hata ile ilgili mesaj verir.

CHKDSK A: ↵

A sürücüsünü kontrol eder. Aşağıdaki mesajı verir. (Sizin disketinize ve disket hatasına göre mesajlarda değişiklikler olabilir.)

Eğer disketimiz hatasız ise:

```
Volume SISTEM DISK created 19/06/1994 23:16
Volume Serial Number is 1C47-1EDA

1.457.664 bytes total disk space
143.872 bytes in 3 hidden files
10.752 bytes in 20 directories
935.936 bytes in 36 user files
367.104 bytes available on disk

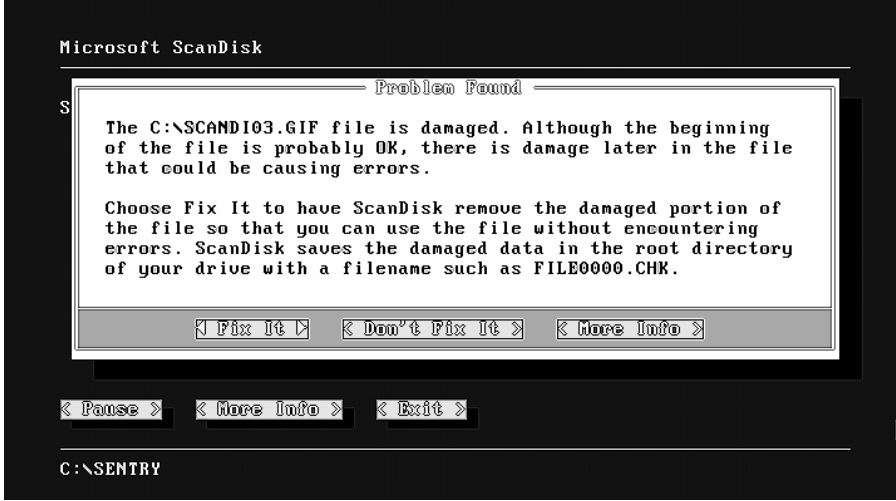
512 bytes in each allocation unit
2.847 total allocation units on disk
717 available allocation units on disk

655.360 total bytes memory
615.312 bytes free

Instead of using CHKDSK, try using SCANDISK. SCANDISK can reliably detect
and fix a much wider range of disk problems. For more information,
type HELP SCANDISK from the command prompt.
```

xxxxxx bytes total disk space	Disketin toplam kapasitesi
xxxxx bytes in x hidden files	Gizli dosya sayısı ve alanı
xxxxxx bytes 12 user files	Disketteki dosya sayısı ve kapladığı alan
xxxxx bytes available on disk	Disketteki boş alan
xxx bytes in each allocation unit	Her sektörün kapasitesi
xxxx total allocation units on disk	Disketteki sektör sayısı
xxxx available allocation units on disk	Kullanılabilir boş sektör sayısı
xxxxxx bytes total memory	Bilgisayarın toplam belleği
xxxxxx bytes free	Bellekteki boş alan

Tarama sırasında bulduğu hataları düzeltme konusunda bizden bilgi isteyecektir. Bilgi isteme penceresi aşağıdaki şekildedir:



Burada gördüğünüz seçeneklerin anlamları:

Fix it	Düzeltmeyi yap.
Don't fix it	Düzeltmeyi yapma.
More information	Ayrıntılı bilgi ver.

Düzeltilme yapma seçeneğini seçtikten sonra, **view log** seçeneği ile diske kayıt edebilirsiniz.

21. KOMUT SATIRINI DÜZENLEME

DOSKEY

Dış Komut

Görevi: Komut satırına girilen verileri belleğinde tutarak, tekrar kullanmayı ve düzenlemeyi sağlar.

En kısa yazım ve kullanım şeklidir. 5.0'ın getirdiği bir yeniliktir. Tampon belleğe **1024 karaktere** kadar girilen verileri aklında tutarak tekrar kullanılması için düzenleme yapar. Edit tuşlarının kullanımını sağlar.

Yazılış:

DOSKEY ↵

Bu yazımı girdiğimizde, **DOSKEY is installed** mesajını alırız.

Tuşlara yüklediği görevler ve kullanılışı:

Tuş	Görevi
↑	Her basışta bir önceki komutu, komut satırına basar.
↓	Her basışta bir sonraki komutu, komut satırına basar.
→	İmleci bir karakter sağa götürür. (Önündeki karakteri silmez.)
←	İmleci bir karakter sola götürür. (Önündeki karakteri silmez.)
Del	İmlecin sağındaki karakteri silerek, diğer karakterleri kendine doğru çeker.
Ins	Insert moduna geçer. Araya karakter ekler.
Home	İmleci satır başına getirir.
End	İmleci satır sonuna götürür.
Esc	Komut satırındaki komutu siler.
Backspace	İmlecin solundaki karakteri silerek, sağındaki karakterleri kendine doğru çeker.
Page Up	Tampon bellekteki ilk komutu komut satırına getirir.
Page Down	Tampon bellekteki son komutu komut satırına getirir.
Ctrl + ←	İmleci bir sözcük sola götürür.
Ctrl + →	İmleci bir sözcük sağa götürür.
Ctrl+Home	İmlecin bulunduğu yerden, satırın başına kadar bulunan karakterleri siler.
Ctrl+End	İmlecin bulunduğu yerden, satırın sonuna kadar bulunan karakterleri siler.
F7	Tampon belleğe, o zamana kadar girilen komutların listesini verir.
Alt+F7	Tampon belleğe, daha önce girilen bütün komutları siler.
F8	Tampon belleğe, daha önce girilen bir komutun ilk harflerini girerek, tamamını yazmayı sağlar.
F9	Tampon belleğe daha önce girilen komutun satır numarası girilerek, yazılmasını sağlar.
Ctrl + T	¶ işaretinin yazılmasını sağlar. İki komut arasına yazılarak, aynı anda iki komutun girilmesi sağlanır.

Diğer fonksiyon tuşlarını daha önceki kullanım biçimi ile burada kullanabilirsiniz. **F3** tampon bellekteki komutu aynen yineler, **F1** karakter karakter yineler gibi...

22. YEDEKLEME

BACKUP

Dış Komut

Görevi: Dosyaların yedeğini almak içindir.

Yazılışı:

```
BACKUP Sürücü1: [\Dizin Adı\] [Dosya Adı] [Sürücü2:]  
[/s] [/m] [/a] [/d:Tarih] [/f:Kapasite] [/l:Dosyaadı]
```

Sürücü1: Yedeği alınacak disk sürücüsü

Sürücü2: Dosyaların yedekleneceği disket sürücüsü

Anahtarlar:

- /s** Disk veya disket üzerinde bulunan dizin ve alt dizinlerin orijinalindeki gibi yedeklenmesini sağlar.
- /m** Bu anahtar ile yapılan en son yedeklemeden sonra, yalnızca üzerinde değişiklik yapılan dosyaları yedekler.
- /a** İlk kez yapılan yedeklemelerde kullanılması gerekli değildir. Yeniden yapılan yedeklemelerde kullanılır. Daha önceden yedeği alınan dosyaların silinmesini engeller. Bu anahtar kullanılmadığı takdirde, daha önce yedeklenen dosyaların silineceği uyarısı, bilgisayar tarafından verilir.
- /d:Tarih** Belirtilen tarihten sonra değiştirilen veya oluşturulan dosyaların yedeğini alır.
- /f:Kapasite** Hedef disk formatlanmamışsa, hedef diskin formatlanması için kapasite belirtilir.
- /l:Dosyaadı** Verilen isimle dosya açılarak, yedekleme ile ilgili bilgiler o dosyada tutulur.

Örnek:

```
BACKUP C:\MUHASEBE\*.DAT A: ↵
```

Yedeklemenin ilk kez yapıldığını varsayarak yazdığımız bu yazılım, hard diskte bulunan MUHASEBE dizininden uzantısı DAT olan bütün dosyaların yedeğini A sürücüsüne alır.

Eğer yedeklenecek dosyalar bir diskete sığmazsa, yedekleme işlemi bitene kadar bizden disket istenir.

23. YEDEKLENMİŞ DOSYALARI GERİ KOPYALAMA

RESTORE

Dış Komut

Görevi: BACKUP ile yedeklenmiş kopyaları, yeniden hard diske kopyalar.

Yazılışı:

```
RESTORE Sürücü1: [Sürücü2:] [\Dizin Adı\Dosya Adı] [/s] [/p]
[/b:Tarih] [/a:Tarih] [/e:Zaman] [/l:Zaman] [/m] [/n] [/d]
```

Sürücü1: Yedeklenmiş dosyaların bulunduğu sürücü.

Sürücü2: Yedeklerin geri kopyalanacağı sürücü.

Anahtarlar:

- | | |
|----------|---|
| /s | Alt dizinleri de kopyalar. |
| /p | En son BACKUP işleminden sonra değişen ve okuma amaçlı dosya uyumu olan dosyaları da kopyalamak için mesaj verir. |
| /b:Tarih | Tarih'ten önce veyaTarih'de değiştirilmiş dosyaları geri kopyalar. |
| /a:Tarih | Tarih'te veyaTarih'ten sonra değiştirilmiş dosyaları geri kopyalar. |
| /m | Son yapılan BACKUP'tan sonra değiştirilmiş dosyaları geri kopyalar. |
| /n | Hedef dosyada bulunmayan dosyaları geri kopyalar. |
| /e:Zaman | Verilen zamandan önce değiştirilmiş dosyaları kopyalar. |
| /l:Zaman | Verilen zamandan sonra değiştirilmiş dosyaları kopyalar. |
| /d | Geri kopyalama işlemi yapmaz. Kaynak sürücüde bulunan ve istenen özelliklere uyan dosyaları listeler. |

```
RESTORE A: C:\MUHASEBE\*. * ↵
```

A sürücüsünde BACKUP'ları alınmış MUHASEBE dizinindeki bütün dosyaları hard diske geri kopyalar.

24. YEDEKLEME ve GERİ YÜKLEME

MSBACKUP

Dış Komut

Görevi: Dosya ya da dosyaların yedeğini alır ve geri yükleme yapar. DOS'un 6.0 versiyonu ile gelmiştir. Menüli çalışma sistemi vardır. Eski versiyonlarda bulunan BACKUP ve RESTORE komutunun görevini yapar. Daha güvenlidir.

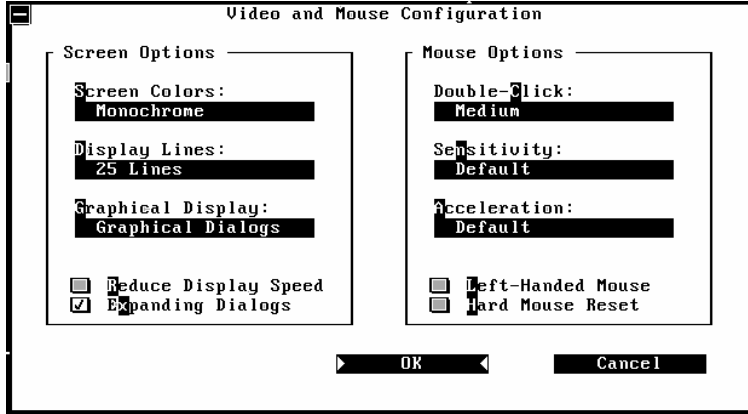
Yazılışı:

MSBACKUP ↵

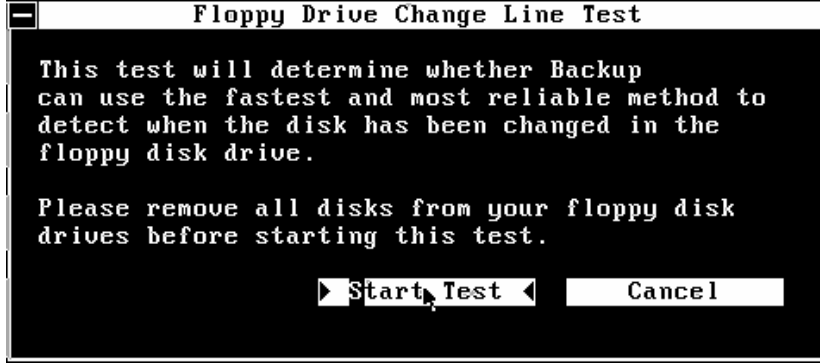
MSBACKUP ilk kez çalıştırılıyorsa aşağıdaki ekran gelerek ayarlama dosyalarını açacak ve gerekli testleri yapacaktır:



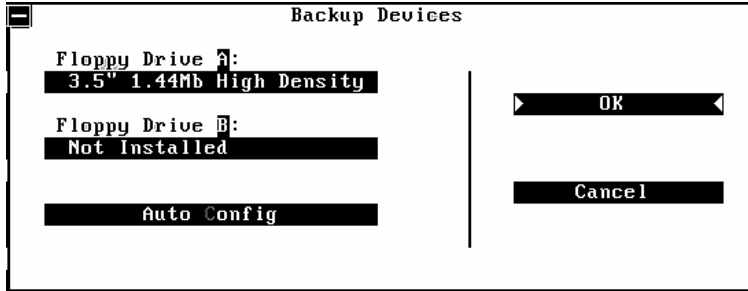
Start Configuration başlamak için, **Quit** çıkış içindir. Geçişleri ok tuşları ve Tab tuşu ile yapabiliriz. Başlama seçildiğinde aşağıdaki ekran gelerek yedekleme aygıtlarının belirtilmesi için giriş yapılması istenecektir:



Bu ekrandan **OK** üzerine gelip **Enter** tuşuna bastıktan sonra teste başlaması için aşağıdaki ekran gelir:



Start Test seçilir ve test başlamış olur. Test sonucunda sürücüler kontrol edilir. Aşağıdaki ekran gelir:



Buradan **OK** seçilerek sürdürülür. Ayarlamalardan sonra artık yedekleme tipi seçilerek yedeklemeye başlanır.

Dosyaları yedekleme:

Backup Type başlığı ile gelen seçeneklerden aşağıdaki yedekleme tiplerinden biri seçilir:

- Full** Tüm dosyaları yedeklemek için.
- Incremental** Son yedeklemeden sonra değişen dosyaların yedeklenmesi için.
- Differential** Son Full yedeklemeden sonra değişen dosyaların yedeklenmesi için.

için.

Backup From	Yedeklemenin hangi sürücü ile yapılacağı belirtilir.
Select Files	Yedekleme yapılacak dosyaların işaretlenerek seçimi sağlanır.
Backup To	Yedeklemenin hangi sürücüye yapılacağı belirtilir.
Option	Yedekleme ile ilgili ayrıntılı düzenleme yapılmasını sağlar.

Yedeklenen dosyaları geri yükleme:

Daha önce yedeklenen dosyaların geri yüklenmesi için, yedekleme penceresinden **Restore** seçeneği seçilir. Geri yükleme sırasında karşımıza gelecek seçeneklerin anlamları aşağıdaki gibidir:

Backup Set Catalog	Backup katalog dosyası belirlenir.
Restore From	Geri yüklemenin nereden yapılacağı belirtilir.
Restore Files	Geri yüklenecek dosyaların seçimi sağlanır.
Select Files	Geri yüklenecek dosyaların işaretlenerek seçimini sağlar.
Restore To	Geri yüklemenin yapılacağı yer belirtilir. Burada karşımıza üç seçenek çıkar: Original Locations Orijinal yerine geri yükleme yapılır. Other Drives Değişik bir sürücüye geri yükleme yapılır. Other Driectories Değişik bir dizine geri yükleme yapılır.
Options	Geri yükleme ile ilgili ayrıntılı düzenleme yapılmasını sağlar.
Catalog	Katalog dosyasının eklenmesini, kopyalanmasını ya da bozulmuşsa onarılmasını sağlar.

Bütün bu işlemlerin yanısıra dosyaları karşılaştırarak, herhangi bir aksaklık olup olmadığı kontrol edilebilir.

25. DOSYA NİTELİĞİNİ DEĞİŞTİRME

ATTRIB

Dış Komut

Görevi: Dosyanın niteliğini gösterir ya da değiştirir.

Yazılışı:

```
ATTRIB [+R|-R] [+A|-A] [+S|-S]  
[+H|-H] [/S] [Sürücü:] [\Dizinadı]
```

Anahtarlar:

- +R Dosyanın niteliğini yalnız okunabilir yapar.
- R Dosyanın yalnız okunabilir niteliğini kaldırır.
- +A Dosyanın niteliğini arşiv dosyası yapar.
- A Dosyanın arşiv niteliğini kaldırır.
- +S Dosyanın niteliğini sistem dosyası yapar.
- S Dosyanın sistem dosyası niteliğini kaldırır.
- +H Dosyanın niteliğini gizli dosya yapar.
- H Dosyanın gizliliğini kaldırır.
- /S Uygulanan komutun alt dizinlerde de geçerli olmasını sağlar.

```
ATTRIB +R AUTOEXEC.BAT ↵
```

AUTOEXEC.BAT dosyasını **Read Only**, yani yalnız okunabilir dosya yapar.

26. DOSYA KARŞILAŞTIRMA

COMP

Dış Komut

Görevi: Aynı uzunluktaki iki dosyayı karşılaştırır. Aralarındaki farkları verir.

Yazılışı:

```
COMP [Sürücü1:] [\Dizin Adı\Dosya Adı1] [Sürücü2:]  
[\Dizin Adı\Dosya Adı2] [/d] [/a] [/l] [/c] [/n=Sayı]
```

Anahtarlar:

- /d Farklılıkları Headecimal olarak gösterir.
- /a Farklılıkları ASCII olarak gösterir.
- /l Farklı karakterlerin bulunduğu satırların numaralarını gösterir.
- /c Karşılaştırma sırasında büyük harf-küçük harf ayrımı yapmaz.
- /n:Sayı Belirtilen sayı kadar satırların karşılaştırmasını yapar.

COMP A:FIRMA.INT C:FIRMA.INT ↵

A sürücüsündeki **FIRMA.INT** adlı dosya, C sürücüsündeki **FIRMA.INT** adlı dosya ile karşılaştırılır.

COMP *.INT *.SPP ↵

Hard diskteki **INT** uzantılı dosyalar ile **SPP** uzantılı dosyaları karşılaştırır.

27. DİSKET KARŞILAŞTIRMA

DISKCOMP Dış Komut

Görevi: İki disketin içeriğini karşılaştırır. Disketlerin içeriklerini (İz ve Sektör) karşılaştırmada kullanılır. Aynı boyut ve kapasitedeki disketleri karşılaştırır.

Yazılışı:
DISKCOMP [Sürücü1:] [Sürücü2:] [/1] [/8]

Sürücü1: Kaynak sürücü,

Sürücü2: Hedef sürücüdür.

Anahtarlar:

- /1** Kullanılan disketin yalnızca bir yüzünü karşılaştırır. Disket iki yüzlü olsa da, yine bir yüzünü karşılaştırır.
- /8** Disketlerde her iz'de 9, 15 ya da 18 sector olsa bile, her iz için 8 sektörü karşılaştırır.

DISKCOMP A: A: ↵

Tek sürücülü bilgisayarlarda bu yazılım kullanılır. Karşılaştırma sırasında kaynak veya hedef disketten hangisini takmanız gerektiği size bildirilecektir.

DISKCOMP A: B: ↵

Çift disket sürücülü bilgisayarlarda, karşılaştırma yapmak istediğimizde bu yazılımı kullanırız. Böyle karşılaştırmalarda, bilgisayar otomatik olarak disketten diskete geçişi sağlayacaktır.

Diskettes compare OK

Karşılaştırma başarı ile sonuçlandırılmış ve iki disket arasında fark yoksa bu mesajı verir.

**Compare error(s) on
Track tt, side s**

Karşılaştırma sonucu, belirtilen track ve yüz açısından farklıdır.

Drive types or diskette types not compatible

İki disket farklıdır. İlk disketin iki yüzü de başarıyla okunmuş fakat disketler birbirinden farklıdır.

Karşılaştırma bitince aşağıdaki mesaj ile karşılaşıyoruz:

Compare another diskette (Y/N)?

Başka bir disket karşılaştırılacak mı? Karşılaştırılacaksa **Y** tuşuna, karşılaştırılmayacaksa **N** tuşuna basarız.

28. VİRÜS İŞLEMLERİ

a. VİRÜS KONTROLÜ

VSAFE

Dış Komut

Görevi: Bu komut bellekte kalıcı bir komuttur. Bellekte tutularak, tanıdığı virüsleri haber vererek kullanıcıyı uyarır.

VSAFE ↵ yazarak belleğe atabilirsiniz. Bu programı AUTOEXEC.BAT dosyasına atarak bilgisayarınızı her açışınızda belleğe yüklü kalmasını sağlayabilir ve programların çalışması sırasında ortaya çıkabilecek virüslere karşı önleminizi almış olursunuz.

Çalıştırdıktan sonra **Alt+V** tuşlarına basarak uyarı seçeneklerini ayarlayabilirsiniz.

b. VİRÜS BULMA, TEMİZLEME

MSAV

Dış Komut

Görevi: Bilgisayarda bulunan, tanıyabildiği virüsleri bulur ve temizler.

Menülü sistemle çalışır. Aşağıdaki yazılış biçimi ve parametrelerini de kullanabilirsiniz:

MSAV

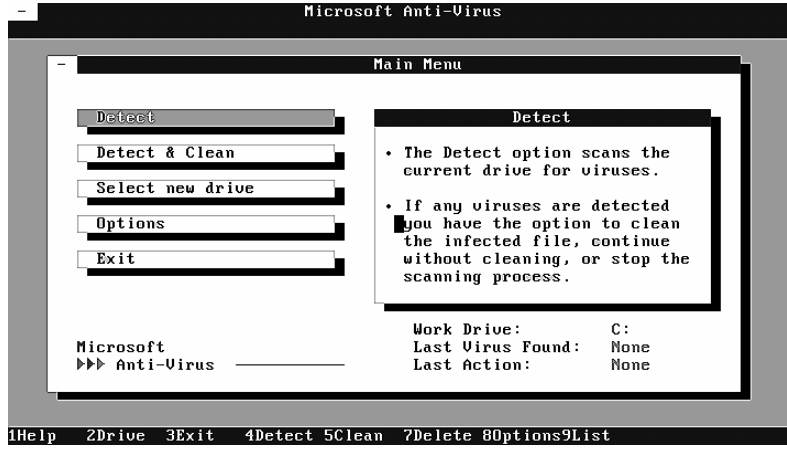
yada

MSAV [Sürücü: [/s] [/c] [/r] [/a] [/l] [/p] [/f]

Anahtarlar:

- /s Virüsü arar, bulur ama temizlemez.
- /c Virüsü arar ve bulunduğunda temizler.
- /r MSAV.RPT adında bir dosya rapor yazar.
- /a Disket (floppy) sürücüler dışındaki tüm sürücülerini tarar.
- /l Network dışındaki tüm sürücülerini tarar.
- /p Menülü ekran çalışması yerine, komut satırından çalışır.
- /f Tarama sırasında dosyaların adlarını görüntülemez.

Menülü sistemde çalışırken aşağıdaki ekran gelir:



Bu ekranda görülen seçimlerin anlamları aşağıdadır:

Detect	Virüsü ara ve bulduğunda size sorar. Virüsü bulduğunda, ya da daha önceki tarama sonrasında dosyalarda bir değişiklik olmuşsa aşağıdaki seçenekler çıkar: Clean: Virüsü temizler. Continiue: Virüsü silmeden taramayı sürdürür. Stop: Arama işlemini sona erdirir. Delete: Virüslü dosyayı disketten siler.
Detect & Clean	Virüsü arar ve bulduğunda temizler.
Select new drive	Yeni sürücü seçmek için.
Options	Virüs arama ile ilgili seçenekleri sunar.
Exit	Programdan çıkış.

Ekranın altında bulunan Fonksiyon tuşlarının görevleri:

F1-Help	Programla ilgili yardım verir.
F2-Drive	Sürücü değiştirmeyi sağlar.
F3-Exit	Programdan çıkış için.
F4-Detect	Virüs arama.
F5-Clean	Virüs arar ve temizler
F7-Delete	Programın virüs ararken yarattığı CHKLIST.MS dosyalarını siler.
F8-Options	Virüs araması için çeşitli seçenekler penceresini getirir.
F9/List	Programın tanıyabildiği virüslerin listesini verir.

29. SİLİNİMİŞ DOSYALARI KURTARMA

UNDELETE

Dış Komut

Görevi: Silinen bir dosyayı yeniden canlandırır.

Yazılışı:

```
UNDELETE [[Sürücü:] [\Dizin]\Dosyaadı] [/dt|/ds|  
/DOS]  
UNDELETE [/LIST|/ALL|/PURGE [Sürücü:] |/STATUS|  
/LOAD|/UNLOAD|/S [Sürücü] |/TSürücü [-Kayıtsayısı]]
```

Anahtarlar:

/LIST	Silinmiş ve kurtarılabilecek dosyaları listeler.
/ALL	Tüm dosyaları size sormadan kurtarır.
/DOS	Yalnızca DOS tarafından silinmiş dosyaları kurtarır.
/dt	Dosya kurtarılmadan önce size sorarak, MIRROR komutunun silme takip dosyasında bulunan dosyaları kurtarır.
/ds	Dosya kurtarılmadan önce onay verilerek, SENTRY dizininde bulunan dosyaları kurtarır.
/LOAD	UNDELETE 'yi belleğe yerleşik (TSR) olarak yükler.
/UNLOAD	Belleğe yerleştirilmiş UNDELETE 'yi yerleşik olmaktan çıkarır.
/PURGE	SENTRY dizinin içeriğini siler.
/STATUS	Silinmiş dosyaların koruma düzeyini gösterir.
/S Sürücü	DELETE SENTRY programını yerleşik olarak belleğe yükler.
/T Sürücü	Delete Tracker korunma düzeyini yerleşik olarak belleğe yükler.
-Kayıtsayısı-	Kayıt sayısı 1-99 arasında olmalıdır.

UNDELETE A:*. * ↵ yazdığımızda.

A sürücüsünde daha önce silinmiş dosya varsa arayarak bunları kurtaracaktır. **DEL** komutu ile silme işleminde, dosyaların adının yalnızca ilk karakteri silinir. **UNDELETE** bu dosyaları bularak yeniden canlandırmayı sağlar.

Aşağıdaki mesajlar çıkar:

```
Directory A:\
File Specifications: *.*

Deletion-tracking file not found.

MS-DOS directory contains xx deleted files .
of those, xx files my be recovered.

Using the MS-DOS directory
```

?xxxx.xxx	xxxxxx	xx-xx-xx	x:xxa	...A
(Dosya Adı)	(Alanı)	(Tarihi)	(Saati)	(Niteliği) Undelete (Y/N)?

Kurtaracağınız bir dosya ise **Y** seçeneğini girin, değilse **N** seçeneğini girin. Şu mesaj belirecektir:

Please type the first character for ?xxxx.xxx: _

Burada sizden dosyanın ilk karakterini girmenizi istemektedir. İlk karakterini anımsamıyorsanız önemli değil. Herhangi bir harf girebilirsiniz. Sonra şu mesaj belirir:

```
File successfully undeleted.  
?xxxx.xxx      xxxxxx      xx-xx-xx      x:xx a      ...A  
Undelete (Y/N)?
```

Kurtarılacak yeni bir dosya varsa bu şekilde sürüp gidecektir.

30. FORMATI GERİ ALMA

UNFORMAT Dış Komut

Görevi: Daha önce formatlanmış bir disketteki bilgileri geri almak için kullanılır. Yalnız disketin DOS'un en az 5.00 uyarlaması ile formatlanmış olması gereklidir.

Yazılışı:

UNFORMAT [Sürücü:] [/u] [/l] [/p] [/j] [/Test] [/Partn]

Anahtarlar:

- /u** Mirror dosyasını kullanmadan diski kurtarır. Mirror programı ile bilgiler daha önce kaydedilmemişse disk tam olarak kurtarılamayabilir.
- /l** Diskteki dosyaların tamamını listeler.
- /p** Ekranı çıkan mesajları yazıcıya aktarır.
- /j** Bu anahtar tek başına kullanılır. Diski kurtarmaz. Mirror programı ile saklanan bilgilerin, disk üzerindeki bilgilerle doğruluğunu kontrol eder.
- /Test** Komutu çalıştırınca ne olacağını göstererek test eder.
- /Partn** Mirror komutu ile /Partn anahtarı kullanılmış ve PARNSAV.FIL dosyasına kaydedilmişse diskin bozulan bölümlerini kurtarır.

UNFORMAT A: ↵

Yazıp ENTER'e bastığımızda şu mesaj belirir:

```
Insert disk to rebuild in drive B:  
and press ENTER when ready.
```

Sonrasında çıkan mesajlara Y ile yanıt verecek olursanız, daha önce formatladığınız disketteki bilgiler kurtarılacaktır.

31. ARAMA YOLU BELİRTME

PATH Komut	İç
----------------------	----

Görevi: Komut arama yolunu ayarlar.

Yazılışı:

PATH [**Sürücü:**] [**Dizin Adı**] [;**Sürücü:**] [**Dizin Adı**] ...]

ya da

PATH;

Yol komutu ile **MS-DOS**'a programların hangi dizinlerden aranması gerektiğini belirtir. Komut arama yolunun uzunluğu en çok **127** karakterdir.

PATH C:\DOS;C:\STOK ↵

MS-DOS ile ilgili dosyaların **DOS** dizininde, **STOK** ile ilgili programların da **STOK** dizininde olduğunu varsayalım. Bu yazılımı **AUTOEXEC.BAT** dosyasına yazacak olursak, bilgisayarı açtığımızda otomatik olarak devreye girecek ve hangi dizinde bulunursak bulunalım bu dosyalarla ilgili işlem yapılacaktır.

32. KOMUT UYARISI

PROMPT İç Komut

Görevi: MS-DOS komut uyarısını değiştirmek için kullanılır.

Yazılışı:

PROMPT [[**Text**] [**\$Karakter**] ...]

Bu komutla **MS-DOS** promptunun değiştirilmesi sağlanır.

<u>Yazım Karakterleri</u>	<u>Çıkacak Karakterler</u>
\$b	Karakteri
\$d	O günkü tarih
\$e	ASCII kod X'1B (Escape)
\$g	> Karakteri
\$h	Backspace'i kullanarak prompt satırına yazılmış bir karakteri silmek için
\$l	< Karakteri
\$n	Varsayılan sürücü

\$p	Üzerinde çalışılan sürücü dizini
\$t	İçinde bulunulan zaman
\$v	Uyarlama (Versiyon) numarası
\$q	= Karakteri
\$\$	\$ karakteri
\$-	ENTER-LINEFEED (Yeni Satır)

Yukarıda verdiğimiz karakterleri yazarak, sağ taraftaki sonuçları alabiliriz.

Örnek:

PROMPT \$P\$G ↵

\$P ile üzerinde çalışılan sürücü, **\$G** ile > karakterini vermesini istediğimizden prompt **A:\>** şeklinde çıkar.

33. BELLEK GÖSTERME

MEM

Dış Komut

Görevi: Kullanılan ve kalan boş belleği ekranda gösterir.

Yazılışı:

MEM [/page] [/classify] [/debug] [/free] [/modülmodüladı]

ya da

MEM [/p] [/c] [/d] [/f] [/m]

Anahtarlar:

/page	ya da	/p	Görüntü bir ekran boyundan büyükse, görüntünün durmasını bekler.
/classify	ya da	/c	Bellekte o an yüklü olan dosyaları ve belleği nasıl kullandığını verir.
/debug	ya da	/d	Bellekteki programların ve dahili sürücülerin listesini verir ve modüller hakkında bilgi verir.
/free	ya da	/f	Konvansiyonel ve üst bellekte boş alanları gösterir.
/modül	ya da	/	Belirtilen programın belleği nasıl kullandığını gösterir. /m anahtarından sonra program adı yazılmalıdır.

MEM/CLASSIFY

Yazarak bellek hakkında bilgi alabiliriz.

MEM ↵ yaptığımızda aşağıdaki ekran gelir.

Memory Type	Total	=	Used	+	Free
Conventional	640K		39K		601K
Upper	91K		91K		0K
Reserved	384K		384K		0K
Extended (XMS)	2.981K		2.981K		0K
Total memory	4.096K		3.495K		601K
Total under 1 MB	731K		130K		601K
Total Expanded (EMS)				1.024K	(1.048.576 bytes)
Free Expanded (EMS)				0K	(0 bytes)
Largest executable program size				601K	(615.088 bytes)
Largest free upper memory block				0K	(0 bytes)
MS-DOS is resident in the high memory area.					

34. BELLEK AYARLAMA

MEMMAKER

Dış Komut

Görevi: Kullanılabilir belleğin daha düzenli çalışması için artırılmış belleği düzenlemek ve bazı programların üst hafızaya atılarak belleğin daha verimli kullanılmasını sağlamak için kullanılır. DOS 6.0 ile gelmiştir.

Komut iletilisinde iken **MEMMAKER** yazarak bellek ayarı programını başlatabilirsiniz. Programı çalıştırdıktan sonra sürücünüzde disket bırakmayınız. Program test ve değişiklik için bilgisayarı kendiliğinden bir kaç kez açıp kapayacaktır.

Ekrana gelen mesaj ile MEMMAKER hakkında bilgi verilir. Devam etmek isteyip istemediğinizi sorar. **Continue** devam etmek için, **Exit** çıkış için kullanılır. Ekrana Continue hazır olarak gelecektir. Değiştirmek için aralık çubuğuna basınız. Sonra istediğiniz seçeneği seçip Enter tuşuna basınız.

Bundan sonra bellek ayarlama için yeni bir ekran gelerek iki seçenek sunacaktır.

Expres Setup: Ayarlamaları kendiliğinden yapacaktır. Optimum seçim sağlayarak yapacaktır.

Custom Setup: Ayarlamaları kullanıcıya sorarak yapar. Bu seçeneği seçerseniz gelen ekrandan istediğinize uygun seçimleri yapmanız gerekir.

Express Setup

Kullanımda yalnızca arttırılmış belleği (expanded) kullanılıp kullanılmayacağı sorulacaktır. Kullanacaksanız Yes seçeneğini seçin, kullanmıyorsanız. No seçeneğini seçin. Bun sonraki sorulara Yes ile yanıt verin.

Custom Setup

Bu seçimde daha öncede belirttiğimiz gibi çeşitli sorularla karşılaşacaksınız. Bu seçeneklerin anlamlarını aşağıda bulacaksınız:

Specify which drives and TSRs to include in optimization?

No'yu seçerseniz tüm donanım ve yerleşik bellek programlarını optimizasyona dahil ederek kurar. **Yes** ile hangi tanım ve programlara işlem yapılacağını siz belirlersiniz.

Scan the upper memory area aggressively?

No'yu seçerseniz EMM386 limitlerinde çalışacaktır. **Yes** seçeneği yüksek bellek alanının kullanılabilir kısmını arttırır.

Optimize upper memory for use with Windows?

Yes Windows altında DOS programları kullanıldığında verimi arttırır. Windows altında DOS programları kullanılmıyorsa **No** seçeneği seçilmelidir.

Use monochrome region (B000-BFFF) for running programs?

No seçeneği monochrom ya da SVGA ekran için, **Yes** seçeneği EGA ve VGA için seçilmelidir.

Keep current EMM386 memory exclusions and inclusions?

EMM386 kullanılıyorsa bazı adresleri dahil ve hariç yapmak için **Yes** seçilir.

Move Expanded BIOS Data Area from conventional to upper memory?

Yes, EMM386'yı üst bellek alanına alarak konvansiyonel belleği arttır. **No** bu işlemi yapmaz.

Bütün bu ayarlamalar yapıldıktan sonra Enter tuşuna basın gelen ekrandan daha önce tanımladığımız seçeneklere göre yanıtları girin. Bu işlemden sonra bilgisayar kendi kendini reset ederek açıp kapanacaktır. Kontrol yapıldıktan sonra gelen ekran için yeniden Enter tuşuna basın. Bilgisayar bir kez daha açılıp kapanacaktır. Hata mesajıyla uyarma yapılmadıysa ekrana önceki ve yeni bellek oranları gelir. Enter tuşuna basılır ve programdan çıkılarak işlem sona erdirilir.

Memmaker'ın yaptığı işlemleri geri almak isterseniz, **MEMMAKER /UNDO** yazıp Enter tuşuna basabilirsiniz.

35. PROGRAMLARI KONVANSİYONEL BELLEĞİN ÜST KISMINA ATMA

LOADFIX
Dış Komut

Görevi: Programları konvansiyonel belleğin 64K'dan daha üst kısma atarak çalıştırır.

Yazılışı:

LOADFIX [Sürücü:][\Dizinadı\Dosyaadı] [Parametreler]

Burada çalıştırılacak programın parametleri varsa onları da kullanabilirsiniz.

LOADFIX C:\DBASE\DBASE ↵

C sürücüsündeki DBASE dizininde bulunan, DBASE adlı programı, 64K'nın üzerine atarak çalıştırır.

36. PROGRAMLARI ÜST BELLEĞE ATMA

LOADHIGH
LH

İç Komut

Görevi: Programları üst belleğe yükler.

Yazılışı:

LOADHIGH [Sürücü:][\Dizinadı\Dosyaadı] [Parametreler]
ya da

LH [Sürücü:][\Dizinadı\Dosyaadı] [Parametreler]

LH C:\DOS\VSAFE

VSAFE adlı program üst bellekte çalışacaktır.

37. DİSK KAPASİTESİNİ ARTTIRMA

DBLSPACE

Dış Komut

Görevi: 6.0'ın getirdiği bir yeniliktir. Diskin kapasitesini ortalama %50 ile %100 arttırır. Bu işlemi diski sıkıştırarak yapar. Bir örnek verecek olursak 170Mb'lık bir disk in kapasitesini 340Mb'a çıkartabilir.

Çalışma sistemi menülüdür. Bu nedenle kolay kullanılır. Komut ileti satırında iken;

DBLSPACE ↵

yazıp Enter tuşuna basmanız programı başlatmak için yeterlidir.

Gelen ekranda **F1** tuşu yardım almak, **F3** tuşu devam etmek ya da çıkmak için kullanılır.

DBLSPACE'in kurulumu da iki yöntemle olur. Hızlı kurma ve sorarak kurma şeklinde:

Express Setup

Hızlı kurmak içindir. Sıkıştırmayı en uygun biçimde yapar. Bu seçim yalnızca **C** sürücüsü için geçerlidir. Başka sürücü için **Custom Setup** seçilmelidir.

Custom Setup

Custom setup seçilerek disk sıkıştırılmasında, sıkıştırmanın nasıl olacağı hakkında size çeşitli sorular yöneltilmektedir.

Gelen ekranda iki seçenek vardır:

Compress an existing drive

Varolan sürücüyü sıkıştırmak içindir. İsteğe göre **C** sürücüsünü içindeki bilgilerle sıkıştırmak için, bu seçeneği kullanabilirsiniz. Bu seçenekte diskin tamamı değilde kullanıcının belirleyeceği kısmı sıkıştırılır.

Create a new empty compressed drive

Yeni bir sıkıştırılmış sürücü yaratır. Bu seçenekte az önceki örneğimize göre **C** sürücüsündeki boş alanı sıkıştırılmış sürücüye dönüştürür. Bu durumda **C** dışında yeni bir sürücü oluşur.

Yeni Sıkıştırılmış Sürücü Yaratma: İkinci seçeneği seçerek yeni bir sıkıştırılmış sürücü yaratılmak istenirse, tüm sürücüler listelenerek çıkacak ekranda bildirilir.

Sıkıştırılmak istenen sürücünün üzerine gelip Enter tuşuna basarak belirtilen sürücüyü sıkıştırabilirsiniz. Sıkıştırma işlemi sırasında bilgisayar iki kez açılıp kapanacaktır.

Disketleri Sıkıştırma: Disketleri sıkıştırmak için aşağıdaki işlemleri sırasıyla yapmamız gerekir:

- ✓ Compress menüsünden Existing Drive komutu seçilir.
- ✓ Gelen ekrandan ok tuşları yardımıyla, sıkıştırılacak olan sürücü işaretlenir ve Enter tuşuna basılır.
- ✓ Devam edilmesi için C tuşuna basılarak sıkıştırma yapılır.

38. DİSK PARÇALANMASINI GİDERME

DEFRAG Dış Komut

Görevi: Diskte oluşan parçalanmayı giderir. Diske kayıt sırasında boş olan sektör- lere kayıt yapılır. Bu da aynı dosyanın, bir parçasının başka sektör- lere kaydı yapıldığından, diskte parçalanmalar oluşur. Bu parçalanmalar hızı yavaşlatır. İşte bütün bunları gidermek için **DEFRAG** komutu kullanılır.

Disk parçalanmasını giderme programını çalıştırmadan önce, gereksiz dosyalar silinir ve tüm programlardan çıkılır.

DEFRAG menü sistemi ile çalışır. Çalıştırmak için **DEFRAG** ↵ yapmanız yeterlidir.

Çalıştırdıktan sonra bilgisayarınıza bağlı olan sürücülerin listesi gelecektir. Burdan sıkıştırmak istediğiniz sürücüyü seçiniz ve **Ok** iletilisinin üzerine geliniz ve **Enter** tuşuna basınız.

Program diskinizi test edip, diskin durumunu kontrol ettikten sonra size öneride bulunacaktır. Enter'a basarak çalıştırmayı yapabileceğiniz gibi **Configuration** menüsünden, istediğiniz seçeneklerle de, çalışmayı yaptırabilirsiniz.

Configuration menüsündeki seçeneklerin anlamları aşağıda verilmiştir:

Begin optimization	Düzenlemeye başlar.
Drive...	Yeni sürücü seçimini sağlar.
Optimization Method...	Düzenlemenin nasıl olacağı belirtilir. İki seçenek çıkar: Full Optimization: Diski en iyi duruma sokar. Tamamını düzenler ve aralarında boşluk bırakmaz. Unfragment Files Only: Yalnızca dosya parçalarını birleştirir. Aralarında boşluk kalabilir.
File sort...	Dosya sıralama yöntemini belirlemenizi sağlar. Aşağıdaki seçenekler çıkar: Unsorted: Sırasız yapar. Name: Dosya adına göre sıralar. Extension: Dosya uzantısına göre sıralar. Date & Time: Kayıt tarihi ve zamanına göre sıralar. Size: Dosyaların kapladığı alana göre sıralar. Ascending: Sıralamaları küçükten büyüğe doğru yapar. Descending: Sıralamaları büyükten küçüğe doğru yapar.
Map legend..	Düzenleme sırasında disk haritasında çıkan sembollerin anlamlarını verir. ■ - Used: Üzerinde bilgi yazılı, kullanılmış blok. ■ - Unused: Boş, kullanılmamış blok. r - Reading: Okuyorum. W - Writing: Yazıyorum. B - Bad: Bozuk blok. X - Unmovable: Taşınamaz blok. Bu tür bloklar Read Only dosyaların yazıldığı bloklardır.
About Defrag...	DEFRAG programı hakkında tanıtım yapar.
eXit	Programdan çıkış.

Yukarıda belirttiğimiz seçimlere göre düzenleme yapılarak disk parçalanması giderilecektir.

39. HARD DİSKİ DÜZENLEME

FDISK

Dış Komut

Görevi: Hard disk MS-DOS'da kullanmak üzere biçimlendirerek düzenler.

Yazılışı:

FDISK [/Status]

Anahtarı:

/Status Hard diskteki bölmelerle ilgili bilgi verir.

Not: Bu komutu kullanırken çok dikkatli olmalısınız. Hard diskinizdeki bilgilerinizi kaybedebilirsiniz.

FDISK yazıp Enter tuşuna bastıktan sonra aşağıdaki ekran gelir:

```
MS-DOS Version 6
                          Fixed Disk Setup Program
                          (C)Copyright Microsoft Corp. 1983 - 1993

                          FDISK Options

Current fixed disk drive: 1

Choose one of the following:

1. Create DOS partition or Logical DOS Drive
2. Set active partition
3. Delete partition or Logical DOS Drive
4. Display partition information

Enter choice: [1]

Press Esc to exit FDISK
```

1. Create DOS partition or Logical DOS Drive: Burada disk seçimi için 1 seçilerek C sürücüsü tanımlanır. DOS için bölme yapılacağı belirtilir. Bu seçenektan sonra aşağıdaki ekran gelir:

```
                Create DOS Partition or Logical DOS Drive

Current fixed disk drive: 1

Choose one of the following:

1. Create Primary DOS Partition
2. Create Extended DOS Partition
3. Create Logical DOS Drive(s) in the Extended DOS Partition

Enter choice: [1]

Press Esc to return to FDISK Options
```

DOS bölümü yaratmak için yine **1.** seçeneği seçerek **Enter** tuşuna basarız. Aşağıdaki ekran gelecektir:

```
                Create Primary DOS Partition

Current fixed disk drive: 1

Do you wish to use the maximum size
for a DOS Portition and make the DOS
portition active (Y/N).....? [Y]

Press ESC the return to Fdisk Options
```

Bu işlemden sonra **Y** tuşuna basılarak DOS bölümü yaratılacaktır.

2. Set active partition: Bu seçim sonrası hard diskte bölünme yapılmışsa, bölümlerle birlikte bilgi verilecektir. Burada hangisi aktif duruma getirilecekse seçim yapılır. Aşağıdaki ekran gelir:

```
                Set Active Partition

Current fixed disk drive: 1

Partition  Status   Type   Volume Label  Mbytes  System  Usage
C: 1       A      PRI DOS  MS-DOS_6      234    FAT16   100%

The only startable partition on Drive 1 is already set active.

Press Esc to continue
```

3. Delete partition or Logical DOS Drive: Bölümleme yapılmışsa bölümlemeyi iptal etmek için seçilir. Aşağıdaki ekran gelir:

```
Delete DOS Partition or Logical DOS Drive

Current fixed disk drive: 1

Choose one of the following:

1. Delete Primary DOS Partition
2. Delete Extended DOS Partition
3. Delete Logical DOS Drive(s) in the Extended DOS Partition
4. Delete Non-DOS Partition

Enter choice: [ ]

Press Esc to return to FDISK Options
```

4. Display partition information: Diskin son durumu hakkında bilgi verir.

```
Display Partition Information

Current fixed disk drive: 1

Partition Status Type Volume Label Mbytes System Usage
C: 1 A PRI DOS MS-DOS_6 234 FAT16 100%

Total disk space is 234 Mbytes (1 Mbyte = 1048576 bytes)

Press Esc to continue
```

40. DİSK CACHE PROGRAMI

SMARTDRV

Dış Komut

Görevi: Yükseltilmiş belleği olan ve hard disk için disk cache programıdır. Hard diskten veri okurken geçen süreyi azaltır.

Yazılışı:

```
SMARTDRV [/x] [/f|n] [/v|q|s] [/c|r] [/l] [/u] [/e:Elmbyk]
          [/b:Arabellek]
```

Anahtarlar:

/x Bütün sürücüler için geri yazma cache belleğini kapatır.
/f Cache'deki bilgiler komut uyarısında görüntülenmeden önce diske yazılır.

/n	Programın otomatik olarak üst belleğe yerleşmesini sağlar.
/v	Durum ve hata mesajlarını görüntüler.
/q	Durum mesajları vermemesini sağlar.
/s	Durum hakkında ek bilgileri getirir.
/c	Cache bilgilerini bellekten cache'e yazar.
/r	Yeniden çalışınca önceki cache bilgilerini siler.
/l	Programın üst belleğe yerleşmesini engeller.
/u	Cache'in CD-ROM sürücüsü için yüklenmesini sağlar.
/e:Elmbyk	Cache belleğin büyüklüğü belirlenir.
/b:Arabellek	Ara belleğin büyüklüğü belirtilir.

41. UYARLAMA TABLOSUNU AYARLAMA

SETVER

Dış Komut

Görevi: Eski versiyonlara ait programları, kendi tablosuna atarak çalıştırılmasını sağlar.

Yazılışı:

```
SETVER [Sürücü:] [\Dizinadı] [\Dosyaadı n.nn]
        [/Delete] [/Quiet]
```

n.nn Çalıştırılacak programın versiyon numarası.

Anahtarları:

/Delete Belirlenen programı uyarlama tablosundan çıkartır.
/Quiet Ekrana mesaj yazılmasını engeller.

DOS'un 6.2 uyarlamasında bulunmayan, ama eski versiyonlarında bulunan komutları bununla çalıştırabiliriz.

```
SETVER EDLIN.EXE 5.00
```

Edlin programı dos'un 6.2 versiyonunda yoktur. Edlin'i çalıştırmak istersek, yukardaki satırı yazarak çalıştırabiliriz.

Tüm eski versiyon komutlarını çalıştırmak için **CONFIG.SYS** dosyasına aşağıdaki komutu ekleyebilirsiniz.

```
DEVICE=C:\DOS\SETVER.EXE
```

MS-DOS artık virüs engelleme, bulma ve temizleme işlemini de yapmaktadır. Bu işlem için komutlar ve görevleri aşağıdadır:

42. VERİ ARAMA

FIND

Dış Komut

Görevi: Bir dosya içinde sözcük ya da belli bir karakter grubunu arar ve ekranda gösterir.

Yazılışı:

```
A>FIND [/v] [/c] [/n] [/i] "String" [[Sürücü:] \DizinAdı  
                                         \Dosya Adı]
```

```
A>FIND "Güven Tanış" MUHASEBE.DAT
```

MUHASEBE.DAT dosyasında **Güven Tanış** yazan satırları gösterir.

Anahtarlar:

- /v Aranan sözcük ya da karakterleri içermeyen tüm satırları gösterir.
- /c Dosyaların içerisinde, uyum gösteren satırları gösterir.
- /n Satırların önlerinde, o satırların numarasını da verir.
- /i Arama yaparken büyük-küçük harf ayrımı yapmadan arar.

43. KLAVYE DEĞİŞTİRME

KEYB

Dış Komut

Görevi: Klavye programı yükler. Çalıştığımız klavyede bulunmayan karakterleri kullanmak gerektiğinde, başka bir klavye yükleyerek istenilen karakterleri kullanmamızı sağlar. Ayrıca tuşların klavye üzerinde dizilişi ülkeye göre değişeceğinden, belirtilen ülkenin standart dizilişinde yükleyecektir.

Yazılışı:

```
KEYB [xx [ , [yyy] , [Sürücü:] [Dizin Adı] ] ] [/id:nnn]
```

xx Ülke kodu (İki harften oluşur.)

yyy Karakter setini belirleyen kod sayfası

Dosya Adı Klavyeyi düzenleyen dosyanın adıdır. Belirtilmezse kullanılan dosya adı **KEYBOARD.SYS**'dir.

/id:nnn Kullanılan klavyeyi tanımlar.

Ülkelerin klavye kodları:

<u>Ülke ve Dil</u>	<u>Klavve Düzeni</u> <u>xx</u>	<u>Klavve Tanımı</u> <u>/ID:nnn</u>	<u>Karakter Seti</u> <u>yyy</u>
Almanya	GR		437,850
Belçika	BE		437,850
Birleşik Amerika	US		437,850
Danimarka	DF		865,850
Finlandiya	SU		437,850
Fransa	FR	120, 189	437,850
Hollanda	NL		437,850
İngiltere	UK	166, 168	437,850
İspanya	SP		437,850
İsveç	SV		437,850
İsviçre, Alman	SG		437,850
İsviçre, Fransız	SF		437,850
İtalya	IT	141, 142	437,850
Kanada-Fransız	CF		863,850
Latin Amerika	LA		437,850
Norveç	NO		865,850
Portekiz	PO		860,850
Türkiye	TR	440, 179	857,850
Yunanistan	GK	319	869, 737
Romanya	RO	333	852,850

KEYB FR ↵

FRANSA klavyesine geçeriz. Değiştirdiğimiz klavye ile çalışırken **CTRL+ALT+F1** tuşlarına basarak klavyeyi eski konumuna geçirebiliriz. Tekrar değiştirdiğimiz klavyeye dönmek istersek **CTRL+ALT+F2** tuşlarına basarak bunu sağlayabiliriz.

Türkiye'nin klavye tanımı dosyası **KEYBRD2.SYS**'dir. Klavye tanımında bu dosyayı kullanmalısınız.

Türkçe klavye ve karakter setini yüklemek için **AUTOEXEC.BAT** dosyasına aşağıdaki komutları yazmalısınız:

```
NLSFUNC
MODE CON CP PREPARE=(857,850) C:\DOS\EGA2.CPI
MODE CON CP SELECT=857
KEYB TR ,,C:\DOS\KEYBRD2.SYS /ID:179
REM /ID:179 TÜRKÇE Q KLAVYE İÇİNDİR.
REM /ID:440 TÜRÇE F KLAVYE İÇİNDİR.
```

CONFIG.SYS dosyasına da aşağıdaki komutları yazınız:

```
COUNTRY=090,857,C:\DOS\COUNTRY.SYS
DEVICE=C:\DOS\DISPLAY.SYS CON=(,2)
```

44. DOSYA SIRALAMA

SORT

Dış Komut

Görevi: Girdiyi okuyarak, bilgiyi sıralar ve sıralanmış bilgiyi ekrana, bir dosyaya veya başka bir çıktı birimine yazar.

Yazılışı:
[Kaynak] | SORT [/r] [+n]
ya da
SORT [/r] [+n] <Kaynak>

Kaynak: Dosya adı veya bir komuttur.

Anahtarlar:

- /r Sıralamayı tersten yapar. Z'den A'ya, 9'dan 0'a doğru sıralar.
+n Dosyayı N satırındaki karaktere göre dizer. Belirtilmezse, dosyayı ilk satırdaki karaktere göre düzenler.

DIR|SORT/+5 ↵

Dosya listesini ilk 5 karakteri gözönüne alarak sıraya dizer ve ekranda gösterir.

SORT<\PASCAL ↵

PASCAL dizinini abece sırasına göre dizerek ekranda görüntüler.

45. GENİŞLETİLMİŞ KARAKTER SETİ

GRAFTABL **Dış Komut**

Görevi: Renkli grafik adaptörü kullanırken (CGA) kullanılabilir, genişletilmiş bir karakter setini ekrana verir. Grafik örneklerini sistem belleğine yükler.

Yazılısı:

GRAFTABL [xxx]

ya da

GRAFTABL/STA [TUS]

xxx Geçerli bir kod sayfasının numarasıdır.

Geçerli kod sayfaları:

<u>xxx</u>	<u>Kod Sayfası</u>
437	Birleşik Devletler
850	Çok Dilli
860	Portekizce
863	Kanada-Fransızca
865	İskandinav

Anahtarlar:

/STA[TUS] Aktif grafik örneğini sisteme yükler.

PRINT GRAFTABL 'ı kullanmak için yardım menüsü ekranda belirir.

GRAFTABL dosyası sistem belleğine yüklendikten sonra aşağıdaki mesaj çıkar:

Graphics characters loaded

Bu mesaj ile kodları 80-FF (Onaltılık) grafik örneklerin de yüklendiğini belirtir.

46. YAZICIYA GRAFİK SETİNİ YÜKLEME

GRAPHICS

Dış Komut

Görevi: Yazıcı üzerine grafik görüntü kartının çizilmesini sağlar. Ekranı yazıcıya gönderirken, grafik varsa yazıcıdan aynen alamayız. Yazıcıya grafik setini yükleyerek ekranın aynen dökümünü alabiliriz.

Yazılışı:

GRAPHICS Type[Profile] [/b] [/r] [/p]

Type Yazıcının özelliklerini belirtir.

Profile Yazıcılardaki bilgileri içeren dosyanın adıdır. Bu dosya adı yazılmazsa **GRAPHICS.PRO** dosyası kullanılır.

Anahtarlar:

/b Renkli yazıcılar içindir. 4 renk ve 8 renk yazıcılar için geçerlidir.

/r Tek şeritli yazıcılar için geçerlidir.

/p İstenilen paralel yazıcının numarası belirtilebilir. Bu değer normalde 0 (Sıfır)'dır.

47. MOD DEĞİŞTİRME

MODE

Dış Komut

Görevi: Çevre birimleri ile iletişimi ayarlar.

Yazılışı: Değişik amaç ve araçlar için değişik yazılımlar gösterir.

Yazıcı Modunun Ayarlanması:

Görevi: Paralel yazıcı için IBM uyumluluk özelliklerini ayarlar.

Yazılışı:

MODE LPTn[:][c][,][1]

ya da

MODE LPTn[Cols=c][Lines=1]

Bu komutun kullanılabilmesi için yazıcının açık olması gerekir. Verdiğimiz komutları yazıcı belleğine alarak uygulamaya koyar.

Anahtarlar:

- n** Yazıcı numarasını belirtir. 1, 2 veya 3 gibi.
c Her satıra yazılacak karakter sayısını belirler. Bu ölçüler 80 ya da 132 olmalıdır. (80 kolonluk yazıcı için)
l Dikey olarak yerleştirilecek satırları ayarlamak için. 6 veya 8 olarak belirlenir. Bu bir inç'e 6 veya 8 satır geleceğini belirtir.

MODE LPT1:132,8

Yukarıdaki örnekte yazıcı kağıda, her satırda **132** karakter ve bir inç'de **8** satır olacak şekilde yazar.

Görüntü Modunun Ayarlanması:

Görevi: Aktif video adaptörü ve görüntü biçimini seçerek monitörü (Ekranı) düzenler.

Yazılışı:

MODE Display,n

ya da

MODE [Display],Shift[,Test]

ya da

MODE Con[:]Cols=m[Lines=n]

Anahtarlar:

- n** Ekran veya görüntü üzerinde, kaç satırlık yer alacağını belirler. Olası değerler 25, 43 ve 50'dir. Ancak her adaptörde uyum nedeniyle bu sonucu vermeyebilir.
m Her satırda kaç karakter olacağını ayarlamak içindir. Olası satır değerleri 40 ve 80'dir.
Display Bu da her satırdaki karakter sayısını belirler. Olası yazım şekilleri şunlardır. 40, 80, BW40, BW80, CO40, CO80, MONO.

40 ve 80 her satırdaki karakter sayısını belirler. **BW** siyah-beyaz, **CO** renkli grafik monitör adaptörünü belirler. **MONO** sürekli her satırda 80 karakterlik bir monochrom görüntü adaptörünü belirler.

- Shift** Renkli Grafik Adaptör görüntüsünün sağa ya da sola çevrileceğini belirtir. Geçerli anahtarlar: L sol için, R sağ için.

- Test** Görüntünün sıraya dizilmesini sağlayarak, ekranın doğru bir biçimde sıralanıp sıralanmadığını da bildirir.
- Con** Hedef sürücüyü tanımlar.

48. MS-DOS EDİTÖRÜ

EDIT

Dış Komut

5.00'ın getirdiği bir yeniliktir. Tam ekran çok rahat bir editör programıdır. Bununla BATCH dosyaları da yazabilirsiniz. Hem klavye hem de mouse ile kullanılabilir. Dosyaları ASCII formatta saklar. Bu dosyaları sonra istediğiniz bir program altında kullanabilirsiniz.

Yazılışı:

```
EDIT [Sürücü:] [\Dizin] [\Dosya Adı.Uzn] [/b] [/g]  
[/h] [/nohi]
```

Anahtarlar:

- /b** Renkli ekran kullanıyorsanız, siyah beyaz ekran biçiminde açılır.
- /g** CGA monitörü için en hızlı ekran yenilemesini sağlar.
- /h** Ekranda olabilecek en çok satırı gösterir.
- /nohi** Ekranı 8 renk kullanmanızı sağlar. (Normalde MS-DOS ekranı 16 renk kullanır.)

EDIT.COM ve **QBASIC.EXE** dosyasını çalışmak istediğimiz sürücüye yükleyerek çalışabiliriz.

EDIT başladığında, ekranın üzerinde **File**, **Edit**, **Search**, **Options** ve **Help** menülerini göreceksiniz. Bu menülere geçmek için **Alt** tuşuna basınız. Hangi menüye geçmek istiyorsanız, **ok** tuşlarıyla menünün üzerine gelip **Enter**'e basınız. Her menünün altında, bir alt menü vardır. Bunları isteğinize göre seçiniz. Eğer yardım almak isterseniz, **F1** tuşu ile anında yardım alabilirsiniz. **EDIT** mouse kullanımını da destekler. Bunların neler olduğunu görelim:



MS-DOS Editör Ekranı

FILE Menüsü:

FILE menüsünün üzerine gelip **Enter**'e bastığımızda, şu menü çıkacaktır:

File	Edit	Search	Options
New			
Open...			
Save			
Save As...			
Print...			
Exit			

- New** Yeni bir dosya yaratmak ve çalışma yapmak için bunu seçeriz. Eğer o an **EDIT**'te bir çalışmamız varsa bu silinerek yeni bir çalışma ortamı yaratılacaktır.
- Open** Daha önce çalıştığımız bir dosyayı açmamız içindir. Yalnız bu dosyaların uzantıları **.TXT** olmalıdır. Biz uzantı vermemişsek, dosya kendiliğinden **.TXT** uzantısını alır. Aradığımız dosya başka bir uzantıda ise ***.TXT** şeklinde çıkan uzantıyı değiştirip istediğimiz uzantıyı yazarak, o uzantıdaki dosyaları bulabiliriz.

- Save** Yazdığımız bilgileri kaydetmek içindir.
- Save As...** Eğer yazdığımız bir bilgiyi başka bir isimle saklayacaksak, **Save As...**'i seçeriz.
- Print...** Dosyaların içeriğini yazıcıdan almamızı sağlar.
- Exit** **EDIT**'ten çıkmamızı sağlar. Eğer, bir çalışma yapmışsak bu çalışmayı kaydedip kaydetmeyeceğimizi de sorar.

EDIT Menü:

Bu menü'yü seçtiğimizde aşağıdaki işlemleri yapabiliriz:

Bu menü ile bloklama işlemlerini yaparız. İstedığımız bir bölümü işaretleyerek silebilir, kesebilir ve başka bir yere kopyalayabiliriz.

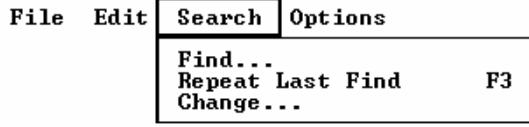
File	Edit	Search	Options
	Cut		Shift+Del
	Copy		Ctrl+Ins
	Paste		Shift+Ins
	Clear		Del

İşaretlemek istediğiniz bölümü bir parmağınız **Shift** tuşuna basılı iken, diğer parmağınızı da ok tuşlarıyla bloklayacağımız bölümün sonuna kadar götürün. Bu işlem zaten renkli bir şerit içinde olduğundan, işaretlenen bölümü rahatlıkla görebileceksiniz.

- Cut** İşaretlenen bölümü kesmek içindir. Bunun için bir parmağınız **Shift** tuşunda iken diğer parmağınızın da **Del** tuşuna basması yeterlidir. İşaretlenen yeri kesecektir.
- Copy** İşaretlenen bölümü kopyalamak içindir. Bunun için de bir parmağınız **Ctrl** tuşunda iken diğer parmağınızla da **Ins** tuşuna basmanız gerekmektedir.
- Paste** Yapıştır anlamındadır. Daha önce **Cut** ile kesilen ya da **Copy** ile kopyalanan bloğu imlecin bulunduğu yere yapıştırır.
- Clear** İşaretlenen bloğu siler.

SEARCH Menü:

Bu bölümde de arama ve değiştirme yapabiliriz. Aranan herhangi bir sözcüğü bulabilir, istersek değiştirebiliriz. Bu bölümü seçtiğimizde alt menü'de şunlar vardır:



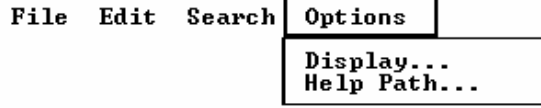
- Find...** Bu bölümde istenen herhangi bir karakter grubu aranabilir. Bunu seçtiğimizde **Find What:** Yazarak bizden aramak istediğimiz sözcüğü girmemizi bekleyecektir. <Ok> yazan yere Tab tuşu ile giderek seçersek, yazdığımızı onaylamış oluruz ve arama başlar. <Cancel> ile istersek buradan çıkabiliriz.
- Repeat Last Find** Bu bölümü seçersek daha önce aradığımız sözcüğü **F3** tuşuna basarak aramanın devamını sağlamış oluruz.
- Change** Bu bölümle belirlenen bir sözcük, istendiğinde başka bir sözcükle değiştirilebilir. **Find What:** Yazan yere değiştirilecek sözcüğü, **Change to:** Yazan yere de yerine gelmesini istediğimiz sözcüğü yazarız.

Change	
Find What:	<input type="text"/>
Change To:	<input type="text"/>
<input type="checkbox"/> Match Upper/Lowercase	<input type="checkbox"/> Whole Word
< Find and Verify > < Change All > < Cancel > < Help >	

Aşağıdaki seçenekleri seçmek için Tab tuşunu kullanabiliriz. <Find and Verify>'i seçersek bulduğu sözcüğü değiştirmeden önce bize soracaktır. Bizim onayımızı aldıktan sonra değiştirme işlemi yapılacaktır. Biz onaylamazsak o yerde değiştirme yapılmayacaktır. <Change All> dosya içerisindeki bütün sözcükleri bize sormadan değiştirecektir. <Cancel> çıkmak için, <Help> yardım almak içindir.

OPTION Menü:

Bu bölümde ekran görünümünü ayarlayabiliriz.



Display... Zemin ve yazı rengini belirler. Scroll çubuğunu açıp kapar. Tab ayarlarını yapar.

HELP Menü:

İstenen herhangi bir konuda yardım almamızı sağlar.

Diğer tuş komutları:

Esc	Komutları keser. Çıkışı sağlar.
Ctrl+A	İmleç bir önceki sözcüğe gider.
Ctrl+C	İmleç bir sayfa aşağı gider. (PgDn)
Ctrl+D	İmleci bir karakter sağa taşır. (Sağ ok tuşu ile aynı görevi yapar.)
Ctrl+F	İmleç bir sonraki sözcüğe gider.
Ctrl+G	İmlecin bulunduğu yerdeki karakteri siler. (DEL)
Ctrl+H	İmleci bir satır yukarı taşır. (Yukarı ok tuşu ile aynı görevi yapar.)
Ctrl+R	İmleç bir sayfa yukarı gider. (PgUp)
Ctrl+S	İmleci bir karakter sola taşır. (Sol ok tuşu ile aynı görevi yapar.)
Ctrl+T	Bir sonraki sözcüğü siler.
Ctrl+V	Insert modunu açar-kapar. Araya karakter girmeye yarar. (INS)
Ctrl+Y	İmlecin bulunduğu satırı siler.
Ctrl+Z	Sayfayı bir satır yukarı kaydırır.
Ctrl+W	Sayfayı bir satır aşağı kaydırır.
Ctrl+X	İmleci bir satır aşağı taşır. (Aşağı ok tuşu ile aynı görevi yapar.)
Ctrl+Q+D	İmleci satırın sonuna götürür.
Ctrl+Q+S	İmleci satırın başına götürür.
Ctrl+PgUp	İmleci bir satır (78 karakter) sağa kaydırır.
Ctrl+PgDn	İmleci bir satır (78 karakter) sola kaydırır.
Home	İmleci satır başına götürür.
End	İmleci satır sonuna götürür.

49. SIKIŞTIRILMIŞ DOSYALARI AÇMA

EXPAND

Dış Komut

Görevi: MS-DOS install disketindeki sıkıştırılmış dosyaları açar. MS-DOS kurma sırasında bu dosyaları kendiliğinden açmaktadır. Bunun dışında gerekli olduğu durumlarda istenen dosyaları açmak için bu komut kullanılır.

Yazılışı:

```
EXPAND [Sürücü1:]\Dizinadı\Dosyaad1  
[Sürücü2:]\Dizinadı\Dosyaad2
```

Sıkıştırılmış dosyalar orijinal diskette _ işareti ile biter.

```
EXPAND A:\FORMAT.CO_ C:\DOS\FORMAT.COM
```

A sürücüsündeki sıkıştırılmış **FORMAT.CO_** dosyasını C sürücüsündeki **DOS** dizinine **FORMAT.COM** olarak açacaktır.

50. SİSTEM KONTROLU

MSD

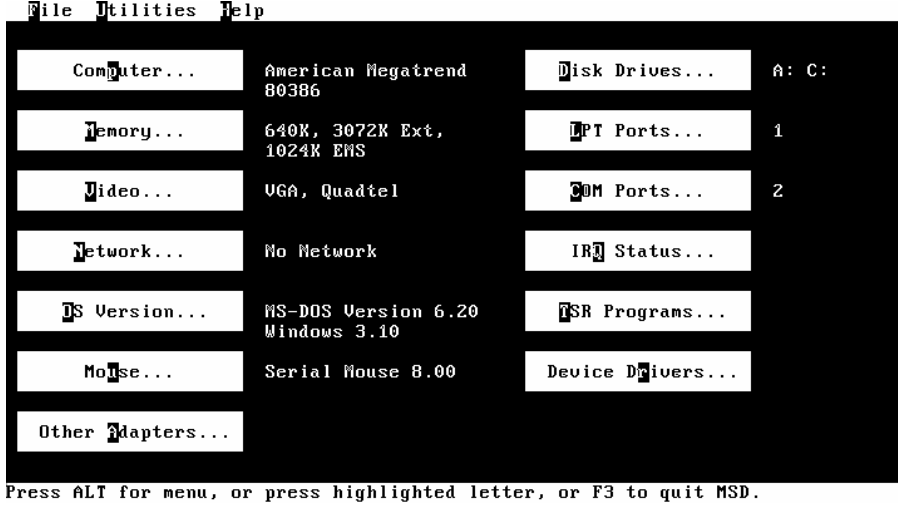
Dış Komut

Görevi: Sistem hakkında bilgi verir.

Menülü sistemle çalışır **MSD** yazıp Enter tuşuna basmak yeterlidir. Aşağıdaki ekran gelerek kontrol yapacaktır:

```
Microsoft (R) Diagnostics  
Version 2.01  
Copyright (C) Microsoft Corporation, 1990-92  
All Rights Reserved  
  
The Microsoft Diagnostics are designed to assist  
Microsoft Product Support personnel in obtaining  
detailed technical information about your computer.  
  
Thank you for using Microsoft Products.  
  
MSD is examining your system ...
```

Kontrol işlemleri bittikten sonra aşağıdaki **Giriş Ekranı** gelecektir:

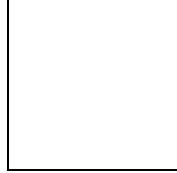


Buradan hangi birim hakkında bilgi almak istiyorsanız onun parlak ışıklı olan harfine bastığımızda ya da mouse ile üzerine gelip tıklattığımızda size o birim ile ilgili bilgi verilecektir.

Ekrandaki birimlerin anlamları şunlardır:

Birim	Tuş	Anlamı
Computer	P	Bilgisayarın ana kartı hakkında bilgi. (Mainboard)
Memory	M	Bellek hakkında.
Video	V	Ekran kartı hakkında.
Network	N	Network ağı hakkında. Varsa.
OS version	O	İşletim sistemi versiyonu.
Mouse	U	Mouse hakkında. Varsa
Other Adapters	A	Diğer adaptörler hakkında.
Disk Drivers	D	Disk sürücülerini hakkında.
LPT Ports	L	Paralel portlar hakkında.
COM Ports	C	Seri iletişim portları hakkında.
TSR Programs	T	Bellekte kalıcı programlar hakkında.
Device Drivers	R	Birim sürücülerini hakkında.

Aşağıdaki ekranda **Computer** bölümü hakkında alınan bilgiyi göreceksiniz:



51. DEBUG

Debug programlarda yanlış bulmak ve düzeltmek test etmek amacıyla kullanılır.

Yazılışı:

DEBUG'u çalıştırmak için DEBUG yazıp Enter'e basarsanız, o ondaki bellek ve disk ile ilgili işlemleri görebilir ve çalışabilirsiniz.

DEBUG Dosyaadı Parametreler

Bu şekilde yazıldığında, istenen dosya ile ilgili işlemler yapılabilir. İsteğe bağlı olarak parametre kullanılabilir.

Debug Komutları:

Debug'un prompt'u (-) işaretidir. Bu işaretin yanına komut yazılır.

- A** Assemble program kodlanması yapılır.
- C** İki bellek bloğu karşılaştırılabilir.
- D** Belleğin içeriğini gösterir.
- E** Belleğin içeriğinin değiştirilmesi sağlanır.
- F** Bellek bloğunun içeriği değiştirilir.
- G** Bellekteki programı çalıştırır.
- H** Onaltılık sistemde toplama ve çıkartma yapılır.
- I** Portlardan değer okunur.
- L** Diskten belleğe bilgi yüklenir.
- M** Bellek bloklarının taşınmasını sağlar.
- N** Dosya adı vermek için.
- O** Portlara değer göndermek için.
- P** Döngü ya da alt programın çalıştırılması.
- Q** Debug programından çıkarak işletim sistemine döner.
- R** Saklayıcı değerlerini görmek ve değiştirmek için.

S	Arama yapmak için.
T	Komutu adım adım çalıştırır.
U	Assembly kodlarının, makine koduna çevrilmesi için.
W	Programı diske kaydetmek için.

G. TOPLU İŞLEM DOSYALARI (BATCH) ve KOMUTLARI

Bilgisayarda çalışma kolaylıklarından biri de **BATCH** dosyalarıdır. Bu tür dosya içinde verilen komutlar, bir veya daha çok dosyaya ulaşmak için yararlıdır. Bir veya daha çok komutu kullanarak **MS-DOS** içinde kısa süre ve yazımla birçok işi bir arada yapmak olanaklıdır.

Toplu işlem dosyalarının uzantısı **.BAT**'tir. Toplu işlem dosyasını çalıştırırken uzantıyı yazmaya gerek yoktur. Yalnızca dosya adını yazıp **ENTER**'e basmak yeterlidir. **BATCH** dosyalarını **EDLIN** ya da **EDIT**'i kullanarak veya dosyaları ASCII kodda kaydedip saklayan herhangi bir kelime işlemci ile yazmak olanaklıdır.

Toplu İşlem Dosyalarını İşletmek İçin Kurallar:

- Toplu işlem dosyasının uzantısı **.BAT** olmalıdır.
- Sürücü adı verilmezse çalıştığımız sürücü, yol adı verilmezse o andaki dizin kullanılır.
- **BATCH** komutları yalnızca, bu tür dosyalar için geçerlidir. **MS-DOS** komutları olarak çalışmaz.
- Çalışılmakta olan **BATCH** dosyasını, **Ctrl+Break** veya **Ctrl+C** ile kesebiliriz. Yalnız böyle bir kesmede **MS-DOS** toplu işlemi sona erdirip erdirmek isteyip istemeyeceğimizi sorar.
- Bir toplu işlem dosyasının içinde son komut olarak başka bir toplu işlem dosyası adı verilebilir.

1. AUTOEXEC.BAT (AÇILIŞ TOPLU İŞLEM DOSYASI)

Bilgisayarı ilk çalıştırdığımızda MS-DOS, takılı olan sürücünün ana dizininde **AUTOEXEC.BAT** adlı dosyayı arar. Eğer bulursa açılıştaki diğer işlemleri bırakarak bu dosyayı işleme sokar. Bu dosyanın mutlaka ana dizinde bulunması gerekmektedir. Değilse bilgisayar açılış sırasında bu dosyayı bulamaz.

Autoexec.bat Dosyasının Yaratılışı:

Bilgisayarın açılışta, istediğimiz şekilde açılmasını istiyorsak, bir **AUTOEXEC.BAT** dosyası yaratmamız gerekir. Diyelim ki açılışta tarih, saat, versiyon verilsin, VSAFE adındaki Virüs Engelleme programı açsın ve istediğimiz klavyeyi yüklesin. Bir de A> üzerinde adımızı yazsın, Promptu da deęiştirsin. Bunun için sırasıyla şu işlemleri yapmamız gerekir:

Açılış toplu işlem dosyasının adı **AUTOEXEC** ve uzantısı **.BAT** olmalıdır.

COPY CON : AUTOEXEC . BAT	
@ECHO OFF	Komutlar çalışırken ekrana yansımaz.
DATE	Tarihi görüntüler, deęiştirmemizi sağlar.
TIME	Saati görüntüler, deęiştirmemizi sağlar.
VSAFE	Virüs kontrol programını açar.
VER	İşletim sisteminin versiyonunu verir.
CLS	Ekranı temizler.
PATH=C : \ ; C : \ DOS	Arama yolunu ayarlar.
PROMPT \$ GÜVEN\$ _ \$P\$G	Promptu deęiştirir.
FKLAVYE	FKLAVYE dosyasını açar.
^Z	Çıkış ve dosyanın diske kaydı için.

Dosyanın diske kaydı için **CONTROL+Z** veya **F6**'ya basarız.

Bilgisayarı yeniden açtığımızda, burada sırasıyla bütün işlemlerin yapıldığını görürüz.

Prompt aşağıdaki şekilde deęişecektir:

```
GÜVEN
A: \>
```

Şeklinde olacaktır. Prompt bu şekilde olursa, bir dizin içerisinde çalışırken dizin adı promptun yanında yazar.

Bir Toplu İşlem Dosyası Yaratmak:

Toplu işlem dosyası yaratırken, açılış toplu işlem dosyası için yazdığımız çoğu kurallar burada da geçerlidir. Burada dikkat edeceğimiz konular şunlardır:

- ⇒ Toplu işlem dosyasının adı, diskette bulunan **.COM** veya **.EXE** uzantılı dosyaların adları veya MS-DOS komutları olmamalıdır.
- ⇒ Dosyanın uzantısı **.BAT** olmalıdır.
- ⇒ Dosya adı için, daha önce geçen kurallara uyulmalıdır.

Örnek:

```
COPY CON:MUHA.BAT
@ECHO OFF
ECHO                MERHABA
ECHO                IYI ÇALIŞMALAR
ECHO                *****
ECHO                HERHANGI BIR TUŞA BASINIZ
PAUSE
MUHASEBE
^Z
```

2. BATCH DOSYASI ALT KOMUTLARI

ECHO

Bir mesaj görüntülemek veya batch komutları ile MS-DOS komutlarının çalışma sırasında görüntülenmesini engellemek için kullanılır.

Yazılışı:

ECHO mesaj	Mesaj görüntülemek için.
ECHO OFF	Diğer BACTH alt komutları veya mesaj komutlarının görüntülenmesini engeller.
ECHO ON	Komut ve mesajların görüntülenmesi için.
@	İşareti ECHO'ların görülmesini engeller.

FOR..IN..DO

Yazılışı:

FOR %%Değişken IN (Set) DO Komut [Parametreler]

Değişken	Tek bir harftir.
Set	Bir veya daha çok sözcük veya dosya terimleridir. Sürücü:\Dizin Adı\Dosya Adı.Uzt şeklindedir. Jokerler kullanılabilir.
Komut	Setteki tüm dosyalar için kullanılacak MS-DOS komutudur.
Parametreler	Kullanılan komutun parametre ya da anahtarları.

GOTO

Toplu işlem dosyasında belirtilen etiketi izleyen satıra atlar ve işlemi bu satırdan sürdürür.

Yazılışı:

GOTO Etiket

Etiket: Önünde iki nokta (:) olan, bir veya daha çok karaktere verilen addır. Etiket adının yalnızca ilk sekiz karakteri geçerlidir.

IF

MS-DOS komutunun koşullu olarak çalışmasını sağlar.

Yazılışı:

IF NOT Koşul Komut

NOT	Koşulun yalnızca sağlanmadığı zaman çalışacağını belirtir.
Koşul	Test edilen şeydir. Karakter dizisi ya da değişken olabilir. A=B gibi... Koşul sağlandığında verilen komut işleyecektir.
Komut	MS-DOS veya BATCH komutudur.

CALL

Toplu işlem dosyası çalışması sırasında başka bir toplu işlem dosyasını çağırır.

Yazılışı:

CALL [Sürücü:] [Dizinadı] [BAT Dosyaadı] [Parametre]

Örnek:

CALL PW.BAT

Toplu işlem dosyası çalışma sırasında, çalışması bitmeden PW.BAT toplu işlem dosyasını çağıracaktır.

CHOICE

Kullanıcıya soru sorularak girilecek yanıtı göre seçim yapılmasını sağlayarak, dosyanın akışını yönlendirir.

Yazılışı:

CHOICE [/C[:]Tuşlar] [/s][/n][/T[:]c,nn] [Metin]

Anahtarlar:

- /C[:]Tuşlar** Uyarı sırasında, yanıt olarak girilecek tuşların tanımı. Tanımlanmazsa **Y** ve **N** olarak atanır.
- /s** Büyük-küçük harf ayırımı yapması sağlanır. Kullanılmazsa büyük küçük harf ayırımı yapmaz.
- /n** Uyarının görüntülenmemesi içindir.
- /T[:]c,nn** Uyarı sırasında giriş yapılması için bekleme süresi ayarlanır.
c: nn ile belirtilen saniye kadar bekledikten sonra giriş yapılmazsa, hazır değeri kullanarak giriş yapılmış gibi işlem sürer.
nn: Bekleme süresi. 0-99 arasında bir değer olmalıdır. 0 verilirse bekleme yapmaz, hazır değeri okur.
- Metin** Yazılacak uyarı metni.

Örnek:

CHOICE /C:EH Evet, Hayır

Bu şekilde yazımdan sonra çalışma sırasında aşağıdaki biçimde görüntü oluşur:

Evet, Hayır [E,H]?

PAUSE

Herhangi bir tuşa basılana kadar toplu işlem dosyası çalışması durur. İsteğe bağlı olarak kullanıcının mesajını görüntüler.

Yazılışı:
PAUSE Mesaj [**>nul**]

Mesaj Görüntülenecek mesajdır.

Bu komuttan sonra **Strike any key when ready ...** mesajı da çıkar.

>nul parametresi kullanılırsa **Strike any key when ready ...** mesajı çıkmaz.

REM

Toplu işlem dosyasına açıklamaları yazmak için kullanılır. Çalışma sırasında görüntülenmez.

Yazılışı:
REM Mesaj

Mesaj Bir satırda en çok 123 karaktere kadar olmalıdır.

SHIFT

Toplu işlem dosyası uyarıldığı zaman, komutta verilen parametreleri bir sola kaydırır.

Yazılışı:
SHIFT

H. KONFIGÜRASYON DOSYASI (DONANIM ÖZELLİKLERİNİ BELİRLEME)

CONFIG.SYS

Görevi: Ünite sürücülere ve çevre ünitelerini kontrol etmek ve kurmak için kullanılır.

CONFIG.SYS dosyasını **ASCII** kaydeden herhangi bir editör (Edlin, Edit vb.) ile yazabilirsiniz. Aşağıda vereceğimiz komut ve anahtarlar yalnızca **CONFIG.SYS** dosyasında kullanılır.

1. KONFIGÜRASYON DOSYASI KOMUTLARI

BREAK

İşletim sisteminin, bir programı durdurmak için, **Ctrl+Break** veya **Ctrl+C**'yi aramasını sağlar.

Yazılışı:

BREAK=ON BREAK'i açmak için.

BREAK=OFF BREAK'i kapatmak için.

BREAK BREAK'in açık mı, kapalı mı olduğunu anlamak için.

BUFFERS

Görevi: Buffer alanını artırarak, harddiske daha hızlı erişimi sağlar.

Yazılışı:

BUFFERS=n, m

n 1'den 99'a kadar belirlenebilen disk buffer'ı sayısı.

m 1-8 arasında giriş-çıkış sırasında yazılabilen ve okunabilen sektör sayısı.
Değer verilmezse 1 kabul edilir.

BUFFERS=20 gibi...

COUNTRY

Görevi: Uluslararası tarih, saat, harf dizimi, para değeri, ondalık ayırımı gibi öğelerin belirlenmesini sağlar.

Yazılışı:

COUNTRY=xxx [, [yyy] [, [Sürücü:] [\Dizinadı\Dosyaadı]]

xxx Ülke kodu.
yyy Kod sayfası.
Dosyaadı Ülke bilgilerinin bulunduğu dosya adı.

Belli başlı ülkelerin kodları ve kod sayfaları:

<u>Ülke va da Dil</u>	<u>Ülke Kodu</u>	<u>Karakter Seti</u>
	<u>xxx</u>	<u>yyy</u>
ABD	001	437,850
Almanya	049	850,437
Belçika	032	850,437
Danimarka	045	850,865
Fransa	033	850,437
Hollanda	031	850,437
İngiltere	044	437,850
İsviçre	041	850,437
İtalya	039	850,437
Türkiye	090	857,850
Uluslararası İngilizce	061	437,850

DEVICE

Görevi: Sistemde kurulabilecek birim sürücülerini kurar.

Yazılışı:

DEVICE=[Sürücü:] [\Dizinadı\Dosyaadı] [Parametre]

Bu komutları kullanırken örneklerimizi, diskinizde DOS dizini olduğunu varsayarak yapacağız. Siz kendinize göre ayarlamayı yapabilirsiniz.

Ünite sürücüsü kurmanın yazılımı:

DEVICE=[Sürücü:] [Dizin Sürücü]

Örnek:
DEVICE=ANSI . SYS
DEVICE=DRIVER . SYS
DEVICE=RAMDRIVE . SYS gibi...

DEVICEHIGH

Görevi: Birim sürücüsünü üst belleğe yükler.

Yazılışı:
DEVICEHIGH=[Sürücü:] [\Dizinadı\Dosyaadı] [Parametreler]

Not: Bu komutu kullanabilmek için **DOS=UMB** komutunu **CONFIG.SYS** dosyasına yazmanız gereklidir.

DOS

Görevi: MS-DOS'un Üst bellekteki bağlantıyı ve üst belleğe yüklenmesini sağlar.

Yazılışı:
DOS=[high|low] , [umb|nomb]

Anahtarlar:
high|low MS-DOS'un bir kısmını üst belleğe yüklemek için **high** parametresini, tamamını yüklemek için **low** parametresi kullanılır.
umb|nomb **umb** parametresi normal bellek ve üst bellek arasında bağlantı oluşturulmasını sağlar, **nomb** bağlantıyı iptal eder.

FILES

Görevi: Birden çok dosyayı kullanıma sunar. Özellikle veritabanı programlarında gerekebilir.

Yazılışı:
FILES=x

x Açık dosya sayısı.

x için değer 8-255 arasındadır. Belirtilmezse 8 kabul edilir.

FILES=20 gibi...

LASTDRIVE

Görevi: Kullanılacak en çok sürücü sayısı belirtilir.

Yazılışı:
LASTDRIVE=x

x A-Z arasında bir harf.

x tanımlaması yapılmazsa E olarak kabul edilir.

MENUCOLOR

Görevi: Açılış menüsünün rengini belirlemeyi sağlar.

Yazılışı:
MENUCOLOR=x, y

x Metin rengini belirler.

y Zemin rengini belirler.

Renk Değerleri:

0	Siyah	6	Kahverengi	11	Parlak cyan
1	Mavi	7	Beyaz	12	Parlak kırmızı
2	Yeşil	8	Gri	13	Parlak magenta
3	Cyan	9	Parlak mavi	14	Sarı
4	Kırmızı	10	Parlak yeşil	15	Parlak beyaz
5	Magenta				

NUMLOCK

Görevi: Bilgisayar açıldığında NUM LOCK'un açık ya da kapalı olmasını sağlar.

Yazılışı:
NUMLOCK=on Açık
NUMLOCK=off Kapalı

REM

Görevi: CONFIG.SYS dosyasına açıklama yazmak için kullanılır.

Yazılışı:

REM Açıklama

REM Bu satır işleme dahil olmayacaktır.

VERIFY

Görevi: Dosyaların diske doğru yazılıp yazılmadığını kontrol eder.

Yazılışı:

VERIFY=on Açık

VERIFY=off Kapalı

2. KURULABİLİR BİRİM SÜRÜCÜLERİ

ANSI.SYS

Görevi: Kullanımı destekleyen ünite sürücüsüdür.

Yazılışı:

DEVICE=[Sürücü:] [Dizin]ANSI.SYS [/x] [/k]

Anahtarlar:

/x 101 tuşlu genişletilmiş klavyelere, tuşların listesini çıkarır.

/k 101 tuşlu genişletilmiş klavyelerde, genişletilmiş tuşları iptal eder.

DBLSPACE.SYS

Görevi: DBLSPACE.BIN'in bellekteki son konumunu belirtir.

Yazılışı:

DEVICE=[Sürücü:] [\Dizinadı]DBLSPACE.SYS [/Move] [/Nohma]

Anahtarları:

/Move DBLSPACE.BIN'i bellekteki son konumuna gönderir.
/Nohma DBLSPACE:BIN'in üst bellekte çalışmasını önler

DISPLAY.SYS

Görevi: Ekran için kod sayfası değişimini sağlar.

Yazılışı:

DEVICE=[Sürücü:] [\Dizinadı]DISPLAY.SYS con[:]=
(Tip[,Kodsy][,n,m])

Anahtarları:

Tip Görüntü adaptörü. MONO, CGA, EGA ve LCD kullanılabilir. Belirtilmezse EGA seçimini yaparak donanımı kontrol eder ve VGA'ya da destekler.

Kodsy Donanımın desteklediği kod sayfasıdır.

Kod Sayfaları:

437 ABD

850 Çok uluslu

857 Türkiye

n Ek kod sayfası. 0-12 arasındaki değerdedir.

m Kod sayfası için alt font sayısı.

DRIVER.SYS

Görevi: Disk sürücünün hazırdaki ölçülerini değiştirerek, yeni değerler verilmesini sağlar.

Yazılışı:

DEVICE=[Sürücü:] [Dizin]DRIVER.SYS [/d:n] [/c] [/f:Faktör]
[/h:Kafa Sayısı] [/s:Sektör] [/t:İz]

Anahtarlar:

/d:n Disk sürücü sayısını belirtir. 0 ile 127 arasındadır.

/c Disk sürücü mandalının açık ya da kapalı olduğunun kontrolünü sağlar.

/f:Faktör Disk sürücüsünün tipini tanımlar.

0 160-180 ya da 320-360K'dır

1 1.2 Mb (5.25 İnce)

2 720 Kb (3.5 İnce)

7 1.44 Mb (3.5 İnce)

9 2.88 Mb (3.5 İnce)

Değer belirtilmemişse ikidir. /F anahtarı kullanıldığında /H, /T ve /S anahtarlarını kullanmaya gerek yoktur.

/h:Kafa Disk sürücüsünün kafa sayısını tanımlar. 1 ile 99 arasındadır. Değer verilmezse 2'dir.
/s:Sektör Bir izdeki sektör sayısını tanımlar. 1 ile 99 arasındadır.
/t:İz Bir yüzdeki iz sayısını belirtir. Tanımlanmamışsa geçerli değer 80'dir. 1 ile 999 arasındadır.

EMM386.EXE

Görevi: Dos uygulamaları için bellekten daha fazla verim alınmasını ve üst belleğe erişimi sağlar. Daha çok 80386 mikro işlemci bir makinede verim artar. 80286 mikro işlemci bir makinede kurmak için, kurulu bir EMS kartı bulunması gerekir.

Yazılışı:

DEVICE=[Sürücü:] [Dizin]EMM386.EXE [on|off|auto] [bellek]

on|off|auto ON açar, OFF kapatır, AUTO otomatik çalıştırma yapar.
bellek EMM386'nın atanmak istendiği bellek miktarı Kb olarak yazılır. Geçerli değerler 16-32768'dir

DEVICE=EMM386.EXE 512 RAM

LIM öykünmesi için 512K RAM'ı bırakacaktır.

HIMEM.SYS

Görevi: High Memory adında bir özel bellek hazırlar. Artırılmış belleği yönetir.

Yazılışı:

DEVICE=[Sürücü:] [\Dizinadı]HIMEM.SYS

PRINTER.SYS

Görevi: Paralel yazıcılar için kod sayfası değişimini sağlar.

Yazılışı:

**DEVICE=[Sürücü:] [\Dizinadı]PRINTER.SYS
lptx=(Tip, [Kodsy], [n])**

Anahtarlar:

Tip	Bağlı yazıcı.
Kodsy	Donanımın desteklediği kod sayfası.
n	Ek kod sayfası.

RAMDRIVE.SYS

Görevi: Belleğin bir bölümünde hayali harddisk yaratır. Daha hızlı bir okuma oluşturur.

Yazılışı:

```
DEVICE=[Sürücü:] [\Dizinadı] [RAMDRIVE.SYS [Büyüklik]
[Sektör] [Giriş] [/e|/a]
```

```
DEVICE=C:\DOS\RAMDRIVE.SYS 640 256 100
```

640Kb kapasiteli **RAM** diski tanımlar ve her sektörü **256byte** uzunluğuna ayarlayarak **100** dizin girişine izin verir.

Anahtarlar:

Büyüklik	Disk büyüklüğü Kb olarak belirtilir. Değer belirtilmezse 64Kb kabul edilir.
Sektör	Sektör büyüklüğü byte olarak belirtilir. Değer belirtilmezse 512 byte kabul edilir.
Giriş	Ana dizin girişi belirtilir. Değer belirtilmezse 64 kabul edilir.
/e	Varsa RAM disk olarak yükseltilmiş belleği kullanmayı sağlar. Bu anahtarla birlikte /a anahtarı kullanılmaz.
/a	Varsa ek olarak yükseltilmiş belleği kullanmayı sağlar. /e anahtarı ile birlikte kullanılmaz.

SETVER.EXE

Görevi: MS-DOS versiyon tablosunu belleğe yükler. MS-DOS'un eski versiyonları versiyon uyumsuzluğu gösterip çalışmayabilir. Çalıştırmak için kullanılır.

Yazılışı:

```
DEVICE=[Sürücü:] [\Dizinadı] SETVER.EXE
```

SMARTDRV.SYS

Görevi: Genişletilmiş ya da uzatılmış bellekte gizli disk oluşturur.

Yazılışı:

DEVICE=[Sürücü:] [Dizin] SMARTDRV.SYS [Başlbüy] [Büy] [/a]

Anahtarlar:

Başlbüy Gizli diskin başlangıç büyüklüğüdür. Değer verilmezse 256'dır. 128 ile 8192 arasındadır.

Büy Gizli disk büyüklüğünü tanımlar.

/a Gizli disk genişletilmiş bellekte oluşur. Belirtilmezse artırılmış bellek kullanılır.

SORULAR

1. Pathname nedir?
2. Dizin (Directory) nedir?
3. Dosya (File) nedir?
4. Dizin ve Dosya arasındaki farklar nelerdir? Açıklayınız.
5. Dosya adı verirken uyulması gereken kurallar nelerdir?
6. * ve ? karakterleri neyi simgeler?
7. İç komut ve dış komut nedir?
8. Formatlama nedir? Sistem formatı nasıl yapılır?
9. Dosya listesini hangi komutla görürüz? Anahtarları nelerdir?
10. Toplu İşlem Dosyası ne demektir? Kurallarını sayınız.
11. CONFIG.SYS dosyasının amacı nedir? Açıklayınız.
12. C sürücüsündeki SEKRETER dizininde bulunan ve uzantısı .BAK olan dosyaları silen komutu yazınız.
13. A sürücüsündeki PW dizininde bulunan tüm dosyaları, C sürücüsündeki PW dizinine kopyalayan komutu yazınız.
14. C sürücüsündeki PW dizininin altına PWDATA dizini açınız.
15. VER ve VOL komutlarının görevleri nelerdir?
16. C sürücüsünde bulunan PW dizini ve PWDATA alt dizini içinde dosyalar vardır. Bu dizinleri dosyaları ile birlikte kaç yöntemle sileriz. Siliniz.
17. Aşağıdaki komutların görevleri nelerdir? Açıklayınız.
 - a. DIR C:\MUHASEBE /S /P /W
 - b. DEL A:\DILLER\COBOL*.COB
 - c. REN C:\PROGRAM\ISLETME1.BAS ISLET.BAS
 - d. FORMAT A:/S/F:720
 - e. MD A:ISCI
 - f. CD WINDOWS\SYSTEM
 - g. COPY A:*. * C:\SEKRETER
 - h. PATH C:\DOS
 - i. TYPE C:\PW\MEKTUP.DAT
 - j. LABEL YAZISMA
 - k. RD A:PW
 - l. XCOPY *. * C:\SEKRETER
 - m. TREE C:\WINDOWS
 - n. COPY *. * C:\DOS

BÖLÜM V.

BELLEK HİZMET PROGRAMI

A. GİRİŞ

Bellek hizmet programları işletim sistemine yardımcı programlardır. İşletim sisteminin yapabildiği bütün işlemleri bellek hizmet programlarıyla daha kolay gerçekleştirebiliriz. Bu nedenle kullanımları yaygındır. PCTOOLS, NORTON gibi bellek hizmet programları en çok kullanılanlardır. Hatta işletim sisteminde yapamayacağımız çoğu şeyi bellek hizmet programlarıyla yapabiliriz.

Bellek hizmet programları, işletim sistemlerinde olmayan bazı özellikleri barındırırlar. Bu yönleriyle MS-DOS işletim sisteminin yeni versiyonlarına bu tür programlardan mantık aktarımı olmuş ve işletim sistemleri de bu doğrultuda kendilerini yenileyip geliştirmişlerdir.

Bu öncülüğe örnek verecek olursak; silinen dosyaları yeniden kullanma mantığı PCTOOLS'dan alınmadır.

Bellek hizmet programlarında komutlar yazılmaz. İşaretlenerek kullanılırlar. Bu nedenle işletim sisteminde kullanılan parametreleri kullanma gibi sorunları da yoktur. Menü ve pencere sistemi ile çalışırlar, mouse destekleri vardır.

B. BELLEK HİZMET PROGRAMLARININ TEMEL MANTIĞI

Bellek hizmet programlarında komutların yazılışını ve parametrelerini ezberlemek ve yazmak zorunda olmadığımızdan işletim sistemlerine göre daha avantajlıdırlar. Kullanımı da işletim sistemlerine göre daha pratik olduğundan çoğunlukla tercih nedenidir. Burada şunu da unutmamak gerekir: Bütün bu özelliklerinin yanısıra işletim sisteminin mantığında da açıkladığımız gibi, işletim sistemine gerek duyarlar.

C. BELLEK HİZMET PROGRAMLARININ TEMEL KAVRAMLARI

Bellek hizmet programlarında kullanılan deyimlerin çoğunu bilgisayar hakkında genel bilgiler ve işletim sistemi konusunda görmüştük. Burada değişiklik gösteren ya da daha önce görmediğimiz kavramları yeri gelince açıklayacağız.

D. PCTOOLS BELLEK HİZMET PROGRAMINA GİRİŞ VE TEMEL AÇIKLAMALAR

Bellek hizmet programlarından söz ederken bu tür programlardan en çok kullanılanlarının **PCTOOLS** ve **NORTON** olduğunu söylemiştik. Biz bu bölümde **PCTOOLS** üzerinde duracağız.

PCTOOLS adını **Personal Computer Tool Services**'den alır. **1985** yılında **Central Point Software** şirketi tarafından ilk versiyonu çıkartılmıştır. Son versiyonları 6.0 ve daha yukarı olanlar, diskte çok yer kaplamaktadır. Daha çok büyük bilgisayarlar ve çok kullanıcı bilgisayarlar için geliştirilmişlerdir. Bizim burada anlatacağımız versiyonu diskette çok rahat çalışabilen ve **170 K**'lık alan kaplayan versiyonudur.

Program iki bölümden oluşmaktadır. Birinci bölüm **Dosya** işlemleri, ikinci bölüm ise **Disk** işlemleridir.

PCTOOLS programını çalıştırabilmek için bir diskete yükleyin ve aşağıdaki şekilde yazarak Enter'e basın.

PCTOOLS ↵

Karşınıza aşağıdaki **Beginning Menü** dediğimiz, açılış menüsü çıkacaktır.

```
WELCOME ?

PC Tools Deluxe R4.21

(C)Copyright 1985,1986,1987,1988 Central Point Software, Inc.
Unauthorized duplication prohibited.

Press any key for File Functions

OR

F3=go directly to Disk and Special Functions
F10=change drive/path from A:\

Press ESC to Exit
```

PCTOOLS Açılış Menüsü

Burada gördüğümüz gibi programın tanıtımı, yazılım şirketi ve hangi tuşlarla neler yapılacağı yazılı.

Press any key for File Funtcions

Dosya işlemleri menüsüne geçmek için herhangi bir tuşa basın.

F3=Go directly to Disk and Special Function

Disk işlemleri menüsüne geçmek için F3 tuşuna basın.

F10=Change drive/path from A:

Çalışma ortamını, alanını, sürücüyü değiştirmek için F10 tuşuna basın.

Press ESC to Exit

Programdan çıkış için ESC tuşuna basın.

Yukarıdaki tuş kullanımlarını içeriden de yaparak aynı işlevleri gerçekleştirebileceğimizden genelde Enter tuşuna basarak dosya işlemleri menüsüne geçeriz.

E. PCTOOLS DOSYA İŞLEMLERİ MENÜSÜNÜN AMACI ve KOMUTLARI

Bir tuşa bastığımızda aşağıdaki ekran çıkacaktır. Bu ekranın nasıl kullanılacağını ekranı inceleyip tanıdıktan sonra göreceğiz.

```
PC Tools Deluxe R4.21                               Vol Label=SISTEM DISK
                                           File Functions                               Scroll Lock OFF
Path=A:\*.*
  Name  Ext  Size Attr  Date      Name  Ext  Size Attr  Date
  IO    SYS  40566 HSR A  9/30/93  LABEL EXE  9390 ...A 9/30/93
  MSDOS SYS  38138 HSR A  9/30/93
  COMMAND COM  54619 ...A 9/30/93
  DBLSPACE BIN  64246 HSR A  9/30/93
  AUTOEXEC BAT   493 ...A 6/01/94
  CONFIG  SYS   284 ...A 6/01/94
  WINA20  386   9349 ...A 8/11/93
  FKLAUYE COM   2486 ...A 1/01/80
  XCOPY  EXE  16930 ...A 9/30/93
  MERTUP  ...A 20713 ...A 12/13/93
  DISKCOPY COM  13335 ...A 9/30/93
  FORMAT  COM  22916 ...A 9/30/93
  CHRDSK  EXE  12241 ...A 9/30/93

14 files LISTed = 305706 bytes. 14 files in sub-dir = 305706 bytes.
0 files SELECTed = 0 bytes. Available on volume = 367104 bytes.

Copy Move cOmp Find Rename Delete Ver view/Edit Attrb Wordp Print List
Sort Help <J>=SELECT F1=UNselect F2=alt dir lst F3=other menu Esc=exit PC Tools
F8=directory LIST argument F9=file SELECTION argument F10=chg drive/path
```


Ekranın tanımı şöyledir:

Vol Label	Disk Etiketi
Scroll lock	Ekranın kaydırma durumu
File Function	Menü Adı (Burada Dosya İşlemleri)
Path	Görüntülenen çalışma ortamı (çalışılan sürücü ve dizin)
Name	Dosya adları
Ext	Dosya uzantıları
Size	Dosyaların kapladığı alan (byte olarak)
Attr	Dosyanın niteliği
Date	Dosyanın yazım tarihi
files LISTed	Listelenen dosya adedi ve kapladıkları alan
files in sub-dir	Alt çalışma alanlarındaki dosya sayısı ve kapladıkları alan
files SELECTed	İşaretli dosya sayısı ve kapladıkları alan
Available on volume	Diskte kalan boş alan

En alt kısımda da komutlar bulunmaktadır. Komutların kullanılacağı harf büyük yazılmış ve renkli ya da parlak olarak ışıklandırılmıştır. Buzı komutların başında da hangi tuşa basılacağı belirtilmiştir.

Örner verecek olursak **Copy** komutunda **C** büyük harfle yazılmış ve ışıklandırılmıştır. Kopya işlemi için **C** tuşuna basılacaktır. **cOmp** komutunda da **O** tuşu büyük yazılmış ve ışıklandırılmıştır. Bu komut için **O** tuşuna basılacaktır. **F10=chg drive/path** yazılımı çalışma ortamı değiştirmek içindir ve kullanmak için **F10** tuşuna basılacaktır.

PCTOOLS programında tuşların görevleri aşağıdaki gibidir:

Scroll Lock OFF durumundayken:

<u>Tuş</u>	<u>Görevi</u>
↓	Bu tuşa her basışta ışıklı gösterge bir alttaki dosya adına gider
↑	Her basışta bir üstteki dosya adına gider.
→	Aynı satırda, sağ taraftaki listeye geçer.
←	Aynı satırda, sol taraftaki listeye geçer.
Home	İşıklı gösterge bulunduğu yerden, ilk sayfanın başındaki dosyanın üzerine gelir.
End	İşıklı gösterge bulunduğu yerden, son sayfanın sonundaki dosyanın üzerine gider.
Page Up	Her basışta önceki 6. dosyaya gider.
Page Down	Her basışta sonraki 6. dosyaya gider.

Scroll Lock ON durumundayken:

<u>Tuş</u>	<u>Görevi</u>
↓	Işıkli gösterge birinci satırda sabit kalırken, dosya adları yukarı doğru kayar.
↑	Işıkli gösterge birinci satırda sabit kalırken, dosya adları aşağı doğru kayar.
→	Işıkli gösterge ikinci bölüme geçer.
←	Işıkli gösterge birinci bölüme geçer.
Home	Işıkli gösterge ilk dosyanın üzerine gider.
End	Işıkli gösterge son dosyanın üzerine gider
Page Up	Işıkli gösterge sabit kalırken, her basışta önceki 6. dosyayı kendine çeker.
Page Down	Işıkli gösterge sabit kalırken, her basışta sonraki 6. dosyayı kendine çeker.

1. DOSYA İŞARETLEME

ENTER Tuşu

Burada görevlerini verdiğimiz edit tuşları ile dosyalar üzerine gidip gelmeyi öğrendik. Komutları kullanırken dosya ya da dosyaları seçmek için ok tuşları ile ışıkli gösterge istenilen dosyanın üzerine getirilir. Birden çok dosya ile ilgili işlem yapacaksak, ışıkli göstergeyi seçtiğimiz dosyanın üzerine getirip **Enter** tuşuna basarız Enter tuşuna bastığımız dosyanın önünde bir numara çıkar. Bunun gibi dosyaları seçtikçe, seçtiğimiz bütün dosyaları numara verilerek işaretlenir ve yapacağımız işlemler o dosyalarla ilgili olur.

Ekranın en altında **↵=SELECT** şeklinde belirtilen komut, dosyaları işaretlemek için yukarıda anlattığımız komuttur.

2. İŞARETLEMEDEN VAZGEÇME

F1=UNselect

İşaretlediğimiz dosya ya da dosyalardan bir yanlışlık sonucu ya da herhangi bir nedenle vazgeçmek istersek **F1** tuşuna basarak, dosyalardaki işaretleri kaldırabiliriz.

3. DOSYA HAKKINDA DETAYLI BİLGİ

F2=alt dir lst

Dosyalar hakkında daha detaylı bilgi almak için **F2** tuşuna basarız. **F2** tuşuna basınca dosyaların **Cluster** numaraları, kayıt saatleri de verilecek ve dosya nitelikleri kısaltma yerine yazıyla bildirilecektir.

4. DISK İŞLEMLERİ MENÜSÜNE GEÇME

F3 other menu

Bulduğumuz menü daha önce de belirttiğimiz gibi, dosya işlemleri içindi. Eğer disk ile ilgili işlemleri yapacaksak, **F3** tuşuna basarak **Disk İşlemleri Menüsü**'ne geçebiliriz.

5. PCTOOLS'TAN ÇIKMA

Esc=exit

PCTOOLS programından çıkmak için **Esc** tuşuna basarız. Ekranda yeni ve küçük bir pencere açılarak bize programdan çıkmak isteyip istemediğimizi soracaktır. **Y** tuşuna basarak programdan çıkabiliriz. Çıkış sırasında aşağıdaki pencere açılacaktır.

Are you SURE you want to
exit PC Tools? (Y/N)

6. AYNI ÖZELLİKTEKİ DOSYALARI GÖRME

F8=directory LIST argument

* ve ? jokerini kullanarak isim benzerliği olan dosyaları **PCTOOLS**'da görebiliriz. Bunu yapabilmek için **F8** tuşuna basarız. **Name** ve **Ext** bölümlerinin karşısında * işaretleri vardır. Bu şekilde **Enter**'le boş geçerse bütün dosyalar görülecektir. Biz burada Adı **CO** ile başlayan ve uzantısı ne olursa olsun bütün dosyaları görelim. Aşağıda yukarıdaki isteğimize göre içi doldurulmuş bir pencereyi görelim.

```
PG Tools Deluxe R4.21                               Vol Label=SISTEM DISK
----- File Functions -----
Path=A:\CO*.*

Enter the Directory List argument.

Enter the name and/or extension to be used as a search
argument for the directory listing. A question mark (?)
is a "WILDCARD" which will always match any character
found in that position. An asterisk in the name or
extension will match that position and all remaining
positions.

      Name=[CO*   ]
      Ext =[*   ]

Name and Extension correct as entered. (Y/N)? [y]

F10 = Blank the Name and Extension. Esc = Return.
```

7. AYNI ÖZELLİKTEKİ DOSYALARI İŞARETLETME

F9=file SELECTION argument

Bulduğumuz sürücü ve dizindeki tüm dosyalarla ilgili işlem yapacağımızda, dosyaları tek tek işaretlemek yerine istediğimiz özellikteki dosyaları ya da tümünü F9 tuşuna basarak işaretleyebiliriz. F9 tuşuna bastığımızda F8'deki gibi bir pencere açılacaktır. * işaretleri kendiliğinden seçilmiştir. Buraya istenen özellikleri belirtebiliriz. Bize Enter tuşuna basarak tüm dosyaları işaretlemek kalıyor. F8'den farkı bütün dosyalar ekranda görünürken istediğimiz özellikte olan dosyaları işaretleyecektir. Burada da dosya adı CO*, uzantısı * şeklinde işaretlediğimizde bütün dosyalar görülecek ama adı CO ile başlayan dosyalar işaretlenmiş olacaktır.

F8 ve F9 arasındaki farka dikkat edersek, F8'de yalnızca özellikli dosyalar görünüyordu. F9'da ise bütün dosyalar görülüyor ama özellikli dosyalar işaretlenmiş biçimde oluyor.

8. SÜRÜCÜ YA DA DİZİN DEĞİŞTİRME

F10=chg drive/path

Çalıştığımız ortamdan diğer bir ortama geçmek için F10 tuşuna basarak istediğimiz sürücü ya da dizini seçebiliriz. F10 tuşuna bastığımızda aşağıdaki gibi küçük bir pencere açılarak hangi sürücüde çalışmak istediğimizi soracaktır.

```
Enter NEW drive letter below. Press
"↵" for no change, "Esc" to return.
NEW Drive ID - [C]
Valid letters are A thru E.
```

Biz burada sürücü değiştirmek için A harfine basalım. Diskete geçişte **Reading system areas.....** Disketi tararken **Scanning sub-driectories..** mesajı çıkacaktır. Eğer disketimizde dizin yoksa hemen ana dizine geçerek dosyaları listeleyecektir. Dizin varsa dizin listesini verecek ve bizim istediğimiz dizine geçerek çalışmamızı sağlayacaktır. İsteddiğimiz dizini seçebilmek için ok tuşları yardımı ile dizinin üzerine gelir ve **Enter** tuşuna basarız. Aşağıda dizin olan bir disketi seçtiğimizde karşımıza çıkacak ekran şeklini görelim:

```
PC Tools Deluxe R4.21                               Vol Label=SISTEM DISK
-----Path Functions-----
Path=A:\                                             BLINK=DOS current
R
D
D
L
|
+---DOS
|   +---PW
|       +---DATA
|           +---OZEL
|               +---ISYAZI
|                   +---KAMU
|                       +---SIRKET
|                   +---KISISEL
|               +---KITAP
|               +---PREMS
|               +---TETRIS
|   +---OYUN
|   +---DILLER
|       +---BASIC
|       +---COBOL
|       +---PASCAL
|   +---WINDOWS
|       +---SYSTEM
|       +---CORELDRM
|       +---WINWORD
|
+-----
Use cursor control keys to follow the chain to the desired directory.
Press ENTER to accept the choice. Press "Esc" to return.
```

9. DOSYA KOPYALAMA

Copy

Dosya ya da dosyaları kopyalayacağımız zaman **COPY** komutu kullanılır. Bu komutu aşağıdaki işlem sırası ile kullanırız.

1. Dosya ya da dosyalar ↵ tuşuna basarak işaretlenir.

2. Copy komutunda daha önce belirttiğimiz gibi C tuşu parlak renkte ve büyük harfle olduğundan C tuşuna basılır. Karşımıza çıkan ekranda aşağıdaki pencere açılır:

```

PC Tools Deluxe R4.21                               Vol Label=SISTEM DISK
-----File Functions-----                       Scroll Lock OFF
Path=A:\*.*
  Name      Ext      Size Attr      Date      Name      Ext      Size Attr      Date
  IO        SYS      40566 HSR A  9/30/93  9 LABEL   EXE      9390 ...A  9/30/93
  MSDOS     SYS      38138 HSR A  9/30/93
  1 COMMAND COM      54619 ...A  9/30/93
  DBLSPACE BIN     64246 HSR A  9/30/93
  2 AUTOEXEC BAT      493 ...A  6/01/94
  3 CONFIG  SYS      284 ...A  6/01/94
  WINA20    386      9349 ...A  8/11/93
  4 FRLAUME COM      1
  5 XCOPY   EXE      1
  MERTUP    2
  6 DISKCOPY COM     1
  7 FORMAT  COM     2
  8 CHKDSK  EXE     1

```

Enter TARGET Drive ID - [C:]
Valid letters are A thru E.
Press ESC to return

```

  14 files LISTed = 305706 bytes.  14 files in sub-dir = 305706 bytes.
   9 files SELECTed = 132694 bytes.  Available on volume = 367104 bytes.

Copy Move cOmp Find Rename Delete Ver view/Edit Attrib Wordp Print List
Sort Help <|=SELECT F1=UNselect F2=alt dir 1st F3=other menu Esc=exit PC Tools
F8=directory LIST argument F9=file SELECTION argument F10=chg drive/path

```

3. Burada kopyalama işleminin hangi sürücüye yapılacağı sorulmaktadır. A tuşuna basıp A sürücüsünü seçerek, kopyalamayı A sürücüsüne yapabiliriz. A tuşuna basınca kopyalama işlemi başlayacak, izin yoksa kopyalama direkt olarak gerçekleşecektir. Eğer belirttiğimiz sürücüde izin varsa, izin listesi çıkar. Hangi dizine kopyalayacaksak, ok tuşları yardımı ile istediğimiz dizini seçer ve o dizin içerisine kopyalamayı sağlarız.

Kopyalayacağımız tarafta aynı adla dosya varsa aşağıdaki ekran çıkacak ve bize bazı sorular soracaktır.

```

PC Tools Deluxe R4.21                               Vol Label=SISTEM DISK
-----File Copy Service-----
Path=A:\
TO Path=C:\
COMMAND .COM is being copied.

File already exists.
Press "R" to replace all files that exist,
      "W" to replace this file only,
      "S" to skip all files that exist,
      "T" to skip this file only,
or Esc to terminate operation.

```

Burada gireceğimiz seçeneklerin anlamları şöyledir:

- “R” R tuşuna basarsak, kopyalama sırasında aynı isimde olan bütün dosyaların üzerine bize bir daha sormadan kopyalama yapacaktır.
- “W” W tuşuna basarsak, yalnız o an ekranda görünen dosyanın üzerine yazar.
- “S” S tuşuna basarsak, aynı isimde olan dosyaları yeniden kopyalamayacak ve atlayarak, orada olmayan dosyaları kopyalayacaktır.
- “T” T tuşuna basarsak, yalnızca ekranda görentülenen dosyayı kopyalamadan atlayacaktır.

10. DOSYA TAŞIMA

Move

Bir ya da daha çok dosyayı bulunduğu ortamdan başka yere taşımak istediğimizde **Move** komutunu kullanırız. Move komutu ile taşıma sırasında, önceki ortamda bulunan dosyalar silinerek yeni ortama taşınacaktır.

Taşıma işlemi için aşağıdaki sıra takip edilir:

1. Taşınacak dosya ya da dosyalar işaretlenir.
2. M tuşuna basılır.
3. Aşağıdaki gibi bir pencere açılır.

```
Please confirm Move operation. It
will DELETE source file(s). Press
"Y" to continue, "Esc" to return.
```

4. Taşımak için Y tuşuna vazgeçmek için ise Esc tuşuna basılır.
5. Bu işlemden sonra hangi ortama taşınacağını soran bir pencere açılır.

```
Enter TARGET Drive ID - [A]
Valid letters are A thru E.
Press ESC to return
```

6. Çıkan pencereye hangi sürücüye taşıyacaksak, o sürücünün adı girilir. Sürücü bulunduktan sonra izin yoksa kopyalama yapılacak, izin varsa izin listesi gelerek bizden izin seçmemizi isteyecektir.

Taşıma olayı bittikten sonra eski ortama dönecektir. Bu ortamda artık işaretlediğimiz dosyalar yoktur. Taşıyacağımız yerde aynı adla başka dosya varsa, kopyalamada olduğu gibi, bize üzerine yazılıp yazılmayacağı sorulacaktır.

11. DOSYA KARŞILAŞTARMA

cOmp

Farklı ortamlarda bulunan bir ya da daha çok dosyayı karşılaştırmak için **cOmp** komutu kullanılır.

Dosya karşılaştırma için aşağıdaki sıra takip edilir:

1. Karşılaştırılacak dosyalar işaretlenir.
2. O tuşuna basılır. Aşağıdaki pencere çıkar:

```
Enter SECOND Drive ID - [A]
Valid letters are A thru E.
Press ESC to return
```

3. Karşılaştıracağımız dosya hangi ortamda ise, o ortamın sürücü harfini gireriz.
4. Karşımıza yeni bir mesaj çıkacaktır:

```
If you want to compare all files with
matching names, press "Y".
Else, press any key to continue.
```

5. Eğer dosyaların adları sorulmadan karşılaştırılması isteniyorsa **Y** tuşuna, değilse herhangi bir tuşa basılır.

Y tuşuna basarsak dosya adları sorulmadan karşılaştırma yapılacaktır. Herhangi bir tuşa basarsak dosya karşılaştırılmadan önce ikinci sürücüdeki dosyanın adı ve uzantısı bize sorulacaktır.

12. KARAKTER GRUBU ARATMAK ve DEĞİŞTİRMEK

Find

Bir ya da daha çok dosya içerisinde herhangi bir karakter grubunu aratmak istediğimizde **Find** komutu kullanılır.

Aratma için aşağıdaki sıra takip edilir:

1. Karakter grubunun aranacağı dosya ya da dosyalar işaretlenir.
2. **F** tuşuna basılır. Karşımıza aşağıdaki ekran çıkar:


```

PC Tools Deluxe R4.21                               Vol Label=SISTEM DISK
----- File Search Service -----
Path=A:\
File=COMMAND.COM

Please enter character string for which to scan below. You may enter the search
string in ASCII or HEX. If you enter HEX, an EXACT match is done. Otherwise a
case-insensitive search is done. You may change entry modes by pressing F1.

[                                                     ] ← ASCII
0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 3 3 3   SEARCH
1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2   ARGUMENT
[                                                     ] ← HEX                                     LENGTH:00

← =begin search  F1=toggle entry mode  ESC=exit
    
```

Burada arayacağımız karakter grubunu iki şekilde girebiliriz. **ASCII** ya da **HEX** yazarak. Köşeli parantezler içine aranacak karakter grubu girilir. **ASCII** değil de **HEX** yazarsak **F1** tuşuna basarak **HEX** yazımına geçebiliriz.

Örnek olarak **COMMAND.COM** dosyasında **Microsoft** sözcüğünü arayalım:

3. **ASCII** yazan yerin karşısına **Microsoft** sözcüğünü girelim. Bu şekilde doldurulmuş bir ekran aşağıdaki şekilde olacaktır:

```

[ M i c r o s o f t ] ← ASCII
0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 3 3 3   SEARCH
1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2   ARGUMENT
14D6963726F736F6674 ] ← HEX                                     LENGTH:09
    
```

Biz **Microsoft** sözcüğünü girerken **HEX** bölümünde de girdiğimiz karakterlerin **HEX** karşılığı yazılacaktır. Yazdıktan sonra **Enter** tuşuna basalım. Arama başlayacaktır ve arama sırasında **Searching sector 0000xx of 0000xxx** yazarak aramanın sürdüğünü bildirecektir. Bulduktan sonra aşağıda;

Press "E" to wiew/Edit the sector or
"G" to continue searching.

mesajı çıkacak. Eğer bulunan karakter grubu için değiştirme yapılacaksa **E** tuşuna, aramaya devam edilecekse **G** tuşuna basmamız gerekir. Aradığımız karakter grubu bulununca **E** tuşuna bastığımızda aşağıdaki ekran çıkacaktır.

```

PC Tools Deluxe R4.21                               Vol Label=SISTEM DISK
-----Sector Edit Service-----
Path=A:\
File=COMMAND.COM   Relative sector 0000016, Clust 00226, Disk Rel Sec 0000257

Displacement  Hex codes  ASCII value
0256(0100)  00 00 00 80 00 00 00 00 00 02 00 46 22 4E 22  Ç      F"N"
0272(0110)  01 2F 43 00 00 00 02 00 46 22 4E 22 01 2F 4D 53  C      F"N"MS
0288(0120)  47 00 00 00 02 00 46 22 4E 22 01 2F 3F 00 00 00  G      F"N"/?
0304(0130)  02 00 46 22 4E 22 01 2F 4B 00 00 00 02 00 46 22  F"N"K      F"
0320(0140)  4E 22 01 2F 59 00 00 00 00 00 00 00 00 00 00 00  N"/Y
0336(0150)  00 00 00 17 49 6E 63 6F 72 72 65 63 74 20 44 4F  #Incorrect DO
0352(0160)  53 20 76 65 72 73 69 6F 6E 0D 0A 1A 4F 75 74 20  S versionOut
0368(0170)  6F 66 20 65 6E 76 69 72 6F 6E 6D 65 6E 74 20 73  of enuironment s
0384(0180)  70 61 63 65 0D 0A 5E 0D 0A 0D 0A 4D 69 63 72 6F  paceJ^JfMicro
0400(0190)  73 6F 66 74 28 52 29 20 4D 53 2D 44 4F 53 28 52  soft(R) MS-DOS(R
0416(01A0)  29 20 56 65 72 73 69 6F 6E 20 36 2E 32 30 0D 0A  ) Version 6.20f
0432(01B0)  20 20 20 20 20 20 20 20 20 20 20 20 20 20 28 43 29  (C)
0448(01C0)  43 6F 70 79 72 69 67 68 74 20 4D 69 63 72 6F 73  Copyright Micros
0464(01D0)  6F 66 74 20 43 6F 72 70 20 31 39 38 31 2D 31 39  oft Corp 1981-19
0480(01E0)  39 33 2E 0D 0A 28 53 70 65 63 69 66 69 65 64 20  93.f(Specified
0496(01F0)  43 4F 4D 4D 41 4E 44 20 73 65 61 72 63 68 20 64  COMMAND search d

↑ ↓ → ← = cursor  F1=swap entry area  F5=update  F6=cancel update  ESC=exit
Home=first pos  End=last pos  PgUp=1st half  PgDn=2nd half
    
```

İmleç Microsoft sözcüğünün M harfinin üzerinde, yalnız HEX bölümündedir.

Yukarıdaki ekranda geçen deyimleri açıklayalım:

File=COMMAND.COM	Dosya adı.
Relative sector	Dosyanın bulunduğu sektör
Clust	Dosyanın bulunduğu blok. (cluster)
Disk Rel Sec	Sektörün diskteki mutlak yeri
Displacement	Bilgilerin bulunduğu konum
Hex codes	ASCII karakterlerin Hexadecimal (16'lık) karşılığı.
ASCII value	Karakterlerin ASCII karşılığı.

Aşağıda belirtilen tuş kullanımlarının görevleri:

cursor	Ok tuşları ile hareket edilerek istenilen karakterin üzerine gelinir.
F1=swap entry area	HEX bölümünden ASCII bölüme, ASCII bölümden HEX bölüme geçiş için.
F5=update	Bir değişiklik yapılmışsa son durumu ile diske kayıt eder.
F6=cancel update	Yapılan değişiklikleri kaydetmeden çıkış.
ESC=exit	Çıkış içindir. Dosya İşlemleri ekranına döner.
Home=first pos	Dosyanın başına gider.
End=last pos	Dosyanın sonuna gider.
PgUp=1st half	Bulduğu sektörün ilk yarısına gider.
PgDn=2nd half	Bulduğu sektörün ikinci yarısına gider.

13. DOSYA ADI DEĞİŞTİRMEK

Rename

Dosya adı değiştirmek için **Rename** seçeneği kullanılır.

Dosya adı değiştirebilmek için sırasıyla aşağıdaki işlemler yapılır:

1. Adı değiştirilmek istenen dosya ya da dosyalar işaretlenir.
2. R tuşuna basılır. [] işaretleri arasında dosya adı ve uzantısı çıkar.

[FKLAVYE]

[COM]

3. Burada yeni vereceğimiz ad girilir **Enter** tuşuna basılır ve aşağıdaki ekran çıkar.

```
FKLAUYE .COM is to be RENAMED to MKLAUYE .COM
Please confirm. "Y" to RENAME
                "N" to reenter
                "B" to bypass
                ESC to return
```

Seçeneklerin anlamları şöyledir:

- “Y” Y tuşuna basarak adı değiştirme işlemi gerçekleştirilir.
- “N” N tuşuna basarak, isim değiştirme ekranına yeniden dönülür ve yeni bir değişiklik yapılmasına izin verir.
- “B” B tuşuna basarak o dosyanın adı değiştirilmeden geçilir.
- ESC Bu ekrandan çıkış içindir.

Yukarıdaki seçeneklere göre işlemimizi yaptıktan sonra, Dosya İşlemleri Menü’süne döner. Yeni adlarıyla dosyaları burada görebiliriz.

Dosya adı değiştirmede **F8** ve **F9**’dan yararlanarak * ve ? joker karakterlerini kullanabiliriz.

14. DOSYA SİLME

Delete

Dosya silme işlemleri için **Delete** seçeneği kullanılır. Burda da **F8** ve **F9**’dan yararlanarak * ve ? joker işaretlerini kullanabiliriz.

Dosya silmek için sırasıyla aşağıdaki işlemler yapılır:

1. Silinecek dosya ya da dosyalar işaretlenir.
2. D tuşuna basılır. Tek dosya sileceksek aşağıdaki seçenekler çıkar:

```
FKLAUYE .COM is to be DELETED
Confirm with "Y" to DELETE
"N" to bypass
Press ESC to Exit
```

Silmek istiyorsak Y tuşuna, silmek istemiyorsak N tuşuna basarız.

Birden çok dosya işaretlemişsek bu kez aşağıdaki seçenekler çıkacaktır.

```
Multiple files were selected to delete.
To delete all files without
individual confirmations, press "Y".
For individual confirmations, press "N".
Press ESC to Exit
```

İşaretlenen dosyaların tamamının sorulmadan, otomatik olarak silinmesini istiyorsanız Y tuşuna, tek tek sorularak silinmesini istiyorsanız N tuşuna basın.

15. DOSYA KONTROLÜ ve DÜZELTME

Ver

Dosyaların kontrol edilmesi ve bozukluk varsa düzeltilebilmesi için Ver, yani Verify seçeneği kullanılır.

Kontrol ve düzeltme işlemi yapabilmek için aşağıdaki sıra takip edilir:

1. Kontrolü yapılacak dosya ya da dosyalar işaretlenir.
2. V tuşuna basılır. Aşağıdaki ekran çıkar.

```
FKLAUYE.COM verifies OK!
XCOPY.EXE verifies OK!
MEKTOP. verifies OK!
DISKCOPY.COM is being verified.      Reading sector 0000000 through 0000007
```

Press ESC to Exit

İşaretlenen dosyaların kontrolü sırasıyla yapılır, bir hata yoksa OK! iletisi ile bildirilir. Dosyaların okunması sektör numaralı ile birlikte gösterilir.

16. DOSYALARIN İÇERİĞİNİ GÖRME ve DÜZELTME

wiew/Edit

Dosyaların içeriğini görüntülemek ve değiştirmek amacıyla **wiew/Edit** seçeneği kullanılır.

Dosya içeriklerini görme ve değiştirme işlemi aşağıdaki sıra ile yapılır:

1. İstenilen dosya ya da dosyalar işaretlenir.
2. E tuşuna basılır ve aşağıdaki ekran gelir.

PC Tools Deluxe R4.21		Vol Label=SISTEM DISK	
File View/Edit Service			
Path=A:*.*			
File=AUTOEXEC.BAT Relative sector 0000000, Clust 00003, Disk Abs Sec 0000034			
Displacement	Hex codes		ASCII value
0000(0000)	40 45 43 48 4F 20 4F 46 46 0D 0A 40 4C 48 20 2F		@ECHO OFF
0016(0010)	4C 3A 30 3B 31 2C 34 35 34 35 36 20 2F 53 20 43		L:0:1,45456 /S C
0032(0020)	3A 5C 44 4F 53 5C 53 4D 41 52 54 44 52 56 2E 45		: \DOS\SMARTDRV.E
0048(0030)	58 45 20 2F 58 0D 0A 50 52 4F 4D 50 54 20 24 20		XE /XJ PROMPT \$
0064(0040)	47 9A 56 45 4E 24 5F 24 70 24 67 0D 0A 43 4C 53		GÜVENŞ_ŞpŞgJCLS
0080(0050)	0D 0A 50 41 54 48 20 43 3A 5C 4D 41 53 41 5C 45		JPATH C:\MASA\N
0096(0060)	58 43 45 4C 3B 43 3A 5C 4D 41 53 41 5C 57 49 4E		XCEL:C:\MASA\WIN
0112(0070)	57 4F 52 44 3B 43 3A 5C 44 4F 53 3B 43 3A 5C 4D		WORD:C:\DOS:C:\M
0128(0080)	4F 55 53 45 3B 43 3A 5C 4D 41 53 41 5C 57 49 4E		OUSE:C:\MASA\WIN
0144(0090)	44 4F 57 53 3B 43 3A 5C 4D 41 53 41 5C 45 58 50		DOWS:C:\MASA\EXP
0160(00A0)	4C 4F 52 45 52 2E 34 4C 0D 0A 53 45 54 20 54 45		LORER.4LJSET TE
0176(00B0)	4D 50 3D 43 3A 5C 44 4F 53 0D 0A 4C 48 20 2F 4C		MP=C:\DOSJLH /L
0192(00C0)	3A 32 2C 36 33 38 34 20 44 4F 53 4B 45 59 0D 0A		:2,6384 DOSKEYJ
0208(00D0)	43 3A 5C 44 4F 53 5C 53 4D 41 52 54 44 52 56 2E		C:\DOS\SMARTDRV.
0224(00E0)	45 58 45 0D 0A 4E 43 43 20 2F 46 41 53 54 4B 45		EXEJNCC /FASTKE
0240(00F0)	59 0D 0A 4C 48 20 2F 4C 3A 32 2C 31 34 33 30 34		YJLH /L:2,14304
Home=beg of file/disk End=end of file/disk			
ESC=Exit PgDn=forward PgUp=back F1=toggle mode F2=chg sector num F3=edit			

Bu ekranda görülen terimlerin anlamları:

File=AUTOEXEC.BAT	Dosya adı.
Relative sector	Dosyanın bulunduğu sektör
Clust	Dosyanın bulunduğu blok. (cluster)
Disk Rel Sec	Sektörün diskteki mutlak yeri
Displacement	Bilgilerin bulunduğu konum
Hex codes	ASCII karakterlerin Hexadecimal (16'lık) karşılığı.
ASCII value	Karakterlerin ASCII karşılığı.

Alt kısımda belirtilen tuş kullanımlarının görevleri:

Home=beg of file/disk	Dosyanın en başına geçer.
End=end of file/disk	Dosyanın en sonuna gider.
PgDn=forward	Görüntüyü yarım sektör ileri alır.
PgUp=back	Görüntüyü yarım sektör geri alır.
F1=toggle mode	Ekranı değiştirerek, yalnızca yazılış durumunu verir.
F2=chg sector num	Ulaşılmak istenen sektör numarasına ulaşım için.
F3=edit	Değişiklik yapmak için HEX bölümünden ASCII bölüme, ASCII bölümden HEX bölüme geçiş.
ESC=exit	Ekrandan çıkış için.

F3 tuşuna basınca ekranın en altındaki tuş kullanımları değişerek aşağıdaki biçimi alır. O dosya içerisinde değişiklik yapmamızı sağlar.

Aşağıdaki tuşların anlamları:

↑ ↓ → ← = cursor F1=swap entry area F5=update F6=cancel update ESC=exit
Home=first pos End=last pos PgUp=1st half PgDn=2nd half

cursor	Ok tuşları ile hareket edilerek istenilen karakterin üzerine gelinir.
F1=swap entry area	HEX bölümünden ASCII bölüme, ASCII bölümden HEX bölüme geçiş için.
F5=update	Herhangi bir değişiklik yapılmış ise son durumu ile disket kayıt eder. Güncelleştirir.
F6=cancel update	Yapılan değişiklikleri kaydetmeden çıkış.
ESC=exit	Çıkış içindir. Dosya İşlemleri ekranına döner.
Home=first pos	Dosyanın başına gider.
End=last pos	Dosyanın sonuna gider.
PgUp=1st half	Bulunduğu sektörün ilk yarısına gider.
PgDn=2nd half	Bulunduğu sektörün ikinci yarısına gider.

17. DOSYALARIN NİTELİĞİNİ GÖRME ve DEĞİŞTİRME

Attrib

Dosyaların niteliğini Attrib komutu ile değiştiririz. Dosya niteliği demek; dosyayı system (sistem), yalnız okunabilir (read only), gizli (hidden), arşiv (archive) demektir.

Dosya niteliğini değiştirme işlemi aşağıdaki sıra ile yapılır:

1. İstenen dosya ya da dosyalar işaretlenir.
2. A tuşuna basılır ve aşağıdaki ekran gelir.

```
PC Tools Deluxe R4.21          File Status Service          Vol Label=SISTEM DISK
Path=A:\
File=AUTOEXEC.BAT           Size =      493 bytes
                             #Clu =        1 clusters

Initial attributes shown are those in effect. To
change, use the arrows (↑↓) to select an
attribute to change. ENTER (↵) will change (toggle) that
attribute. Pressing "U" will update and make the changes
permanent. "ESC" will return without any change.

Initial Attributes          New Attributes
Read Only - OFF            Read Only - OFF
Hidden - OFF                Hidden - OFF
System - OFF                System - OFF
Archive - ON                Archive - ON

Initial Time/Date          New Time/Date
11:12p                      11:12p
06/01/94                    06/01/94

"U" to update and make changes permanent. ESC to return without changes.
← = prior entry → = next entry
```

Ekranda görünen terimlerin anlamları:

Size	Dosyanın kapladığı alan. (byte olarak)
#Clu	Cluster numarası
Initial attributes	Hazır olan, önceden verilen nitelikler.
New attributes	Değiştirilecek ya da değiştirilen nitelikler.
Read Only	Yalnız okunabilir nitelik.
Hidden	Gizli dosya niteliği. (DIR komutu tek kullanıldığında görünmez.)
System	İşletim sistemi programı
Archive	Arşiv dosya
Initial Time/Date	Daha önce kayıt edilmiş, tarih ve saat.
New Time/Date	Değiştirilecek tarih ve saat.

Tuşların görevleri:

←=prior entry	Dosyanın önceki özelliklerine geçmek için.
→=next entry	Dosyanın yeni özelliklerine geçmek için.
U	Yapılan değişiklikleri güncelleştirerek, diske kayıt eder.

3. Değiştirilmek istenen özelliğin üzerine ışıklı gösterge ok tuşları yardımı ile getirilir. Enter tuşuna basılarak OFF ise ON, ON ise OFF yapılır.

OFF	Kapalı. Yani dosyanın o niteliği yok.
ON	Açık. Dosyanın belirlenen özelliği var.

4. Güncelleştirmek için U tuşuna basılır.

18. BELGE DÜZENLEYİCİ (KELİME İŞLEM, EDITOR) KULLANIMI

Wordp (WORDPROCESSOR)

PCTOOLS'un çok yönlü bir program olduğunu buraya kadar öğrendiğimiz bilgilerden anlamışsınız. Bütün bu yeteneklerinin yanısıra bir kelime işlem bölümü de bulunmaktadır.

Burada öğreneceğimiz kelime işlem bölümü ile BATCH dosyaları yazabilir ya da daha önce yazdığımız bir dosya üzerinde düzenlemeler yapabilirsiniz.

PCTOOLS'un kelime işlem bölümü olan **Wordp**'yi kullanabilmek için hangi dosya ile çalışacaksınız, ışıklı göstergiyi o dosyanın üzerine getirip, **W** tuşuna basmanız gerekmektedir. Bundan sonra karşımıza ekranın altında şöyle bir mesaj çıkacaktır.

Press F2 to create a new file, Esc to end,
or any other key to edit XXXXXXXX.XXX

Burada söylenmek istenin, yeni bir dosya yaratacaksanız **F2** tuşuna, işaretli dosya ile çalışacaksınız herhangi bir tuşa, çıkış için **Esc** tuşuna basın anlamındadır.

Aşağıda **F2**'ye basarak yeni yaratılmış bir kelime işlem ekranını görelim:

```
PC Tools Deluxe R4.21                               Vol Label=SYSTEM DISK
-----Word Processing Service-----
Path=A:\
x
1 10 20 30 40 50 60 70 80
  <F2> <F3> <F4> <F5> <F6> <F7> <F8> <F10> INSERT
  Save Search Replace Select Cut Copy Paste show NUM LOCK
```


a. Yazılan Metnin Diske Kaydı (Save)

Bir metni yazdıktan sonra diske kaydetmek için **F2** tuşuna basılır ve dosya yeni ise aşağıdaki mesaj çıkararak dosyaya bir isim vermemizi ister.

```
Please enter the new file and extension names. (ESC will end).
      Name=[      ]
      Ext =[    ]
```

Name'in karşısına dosya adı, **Ext**'in karşısına da dosya uzantısı yazılır ve **Enter**'e basılır. Vazgeçmek için **Esc** tuşuna basılır.

Daha önce yazdığımız bir dosya ise kayıt edecek ve aşağıda;

```
File saved successfully.
Press any key to continue
```

ileti görünerek bir tuşa basmamızı isteyecektir. Herhangi bir tuşa basınca ekran eski konumu alacaktır.

Dosyadan çıkarken son şeklini kaydetmemişsek aşağıdaki mesajla bizi uyararak, son şeklini kaydetmediğimizi, kaydedip kaydetmeyeceğimizi sorar:

```
You have not saved the file since last changing it.
Do you want to save it before exiting? Reply "Y" to save, or "N"
to exit WITHOUT saving the file (Esc returns to word processor).
```

Kaydetmek istiyorsak **Y** tuşuna, kayıt etmeden çıkmak istiyorsak **N** tuşuna basarız.

b. Bir Karakter Grubunu Aratma (Search)

Yazı içerisinde özellikle bulmak istediğimiz bir karakter grubu olabilir. Böyle durumlarda biz istediğimiz sözcüğü yazarak, kelime işlem programımızdan bunu bulmasını isteyebiliriz.

Bunu yapabilmek için **F3** tuşuna basarız ve karşımıza ekranın altında aşağıdaki ileti çıkar:

```
Please enter characters to search for below:
      [                                ]
```

Aratmak istediğimiz karakter grubunu [] işaretleri arasına girerek aratırız. Aynı karakter grubu birden çok ve bunları da bulmak istiyorsak her biri için **F3** tuşuna basarız. Her basışta bir sonrakini bulur.

c. Bir Karakter Grubunu, Bir Karakter Grubu ile Değiştirmek (Replace)

Yazı içerisinde bulunan bir karakter grubunu, diğer bir karakter grubu ile değiştirmek isteyebiliriz. Bunu yapabilmek için **F4** tuşuna basarız ve ekranın altında aşağıdaki ileti çıkar:

```
Please enter characters to replace below:
[ ]
```

Buraya aratmak istediğimiz karakter grubunu yazar ve **Enter** tuşuna basarız. Hemen arkasından bu bölümde bir değişiklik olarak şu biçimi alır:

```
Please enter characters to replace with below:
Replacing:
[File ] [Dosya ]
```

Buradaki örneğimizde içi dolu bir değiştirme işlemi görüyorsunuz. **File** sözcüğünü **Dosya** sözcüğü ile değiştirmek istiyoruz. Yani ilk bölüme aranan sözcük, ikinci bölüme yerine geçmesi istenen sözcük yazıldı.

Değiştirmek istediğimiz sözcük birden çok ise **F4** tuşuna her basışımızda bir sonraki sözcüğü değiştirecektir. Böyle durumda aşağıdaki ileti çıkacaktır.

```
Argument found; press F4 to repeat replace, or Esc to end replace mode.
Replacing: Replacing: with:
[File ] [Dosya ]
```

Değiştirme işlemi sona erdirip çıkmak için **Esc** tuşuna basarız.

d. Bir bölümün İşaretlenerek Bloklanması (Select)

Bir bölümü işaretleyip bloklamak için, işaretlemek istediğimiz yerin başına gelir ve **F5** tuşuna basarız. **F5** tuşuna basmakla başlangıç yerini işaretlemiş olduk. Ok tuşları yardımıyla işaretleyeceğimiz yerin sonuna kadar gideriz. Göreceğiz ki, işaretlediğimiz yer ekranın özelliğine göre renk değiştirecek ya da ışıklandırılacaktır.

Artık bu bölümü ister sileriz, ister kopyalarız. Silmek için **F6** (Cut)tuşuna, kopyalamak için **F7** (Copy)tuşuna basarız. Sildiğimiz ya da kopyaladığımız bölümü başka bir yere yeniden yazmak istersek, imleci yazmak istediğimiz yere getirir ve **F8** (Paste)tuşuna basarız.

19. YAZICIYA YAZDIRMA

Print

İşaretlenmiş dosya ya da dosyaların yazıcıdan yazdırılması **Print** seçeneği ile yapılır.

Yazıcıya yazdırmak için aşağıdaki sıra takip edilmelidir:

1. Yazdırılacak dosya ya da dosyalar işaretlenir.
2. P tuşuna basılır. Bu seçimden sonra aşağıdaki ekran gelir.

```
PC Tools Deluxe R4.21          File Print Service          Vol Label=SISTEM DISK
Path=A:\
File=AUTOEXEC.BAT
Please specify your print options as follows:
"P" - print as a standard text file <file contains control characters
      as necessary>
"W" - print as a standard text file <using PC Tools print options>
"D" - each sector DUMPED in ASCII and HEX
"N" - skip this file and go to next one selected
ESC - return
```

Buradaki seçeneklerin anlamları şöyledir:

- “P” Normal yazım şeklidir. Kontrol karakterleri istenirse verilir.
- “W” Normal yazım şeklidir. PCTOOLS’un seçenekleri kullanılarak basım sağlanır.
- “D” Sektörlerin ASCII ve HEX kodlu olarak yazımı sağlanır.
- “N” İşaretlenen dosyalardan o an ekranda görünen dosyanın yazılmadan atlanması isteniyorsa bu seçenek kullanılır.
- “ESC” Geri dönüş için.

P seçeneği ile yazım şekli:

Bu seçenek kullanıldığında normal yazım şeklinde yazım yapılacaktır. Printer hazır değilse PCTOOLS tarafından uyarılacaktır.

Yazım sırasında aşağıdaki mesaj çıkacaktır:

```
Printing in progress
Press ESC to Exit
```

Yazdırma işlemi başlamıştır. Vazgeçmek için **Esc** tuşuna basılır.

W seçeneği ile yazım şekli:

Bu seçenek de normal yazım şeklidir. Yalnız yazım başlamadan önce sayfa düzeni ve yazıcı tanımlaması için aşağıdaki ekran gelir:

```
PC Tools Deluxe R4.21          File Print Service          Vol Label=SISTEM DISK
Path=A:\
File=AUTOEXEC.BAT

To modify the print options, use the arrows to select an option that
you want to change. Then press the ENTER (↵) key. You will now see a
cursor. Key in the new value and press ENTER (↵) to make the change
final. The cursor will then go away.

When you have no more changes to make, select the last option and press
ENTER (↵). Press Esc to exit.

Lines per page          66      Page headings (Y or N)      N
Margin lines top and bottom  4      Page footings (Y or N)     N
Extra spaces between lines  0      Page numbers (Y or N)      N
Left margin             1      Want to stop between pages? N
Right margin            080     Want to eject last page?   Y
The line length is 080 characters | Printer options are correct

Number of actual lines on a page (including top/bottom margins).
```

Burada belirtilen yazdırma seçeneklerini değiştirmek için imleç ok tuşları yardımı ile istenilen seçeneğin üzerine getirilir **Enter** tuşuna basılır. İstenilen değişiklik yapıldıktan sonra **Enter** tuşuna bir kez daha basılır.

Burada geçen seçeneklerin anlamları şöyledir:

Line per page	Bir sayfadaki satır sayısı.
Margin lines top and bottom	Sayfa üstünde ve altında boş bırakılacak satır sayısı.
Extra spaces between lines	Satırlar arasında bırakılacak boş satır sayısı.
Left margin	Kağıdın sol tarafında kaç karakterlik boşluk bırakılacağı.
Right margin	Kağıdın sağ tarafında kaç karakterlik boşluk bırakılacağı.
The line length is 080 characters	Bir satıra yazılabilecek karakter sayısı.
Page headings (Y or N)	Sayfaya başlık verdirilip verdirilmeyeceği.
Page footings (Y or N)	Sayfa sonuna dipnot verilip verilmeyeceği.
Page numbers (Y or N)	Sayfalara numara verilip verilmeyeceği.
Want to stop between pages?	Sayfa arasında bekleme yapılıp yapılmayacağı.
Want to eject last page	En son sayfadan sonra sayfa başı yapılıp yapılmayacağı.
Printer options are correct	Yazıcı tanımlamalarının doğru olup olmadığı.

Yukarıdaki seçenekler kullanılıncaya yeni ekran gelecek seçeneklerin kullanımı aşağıdaki gibidir:

Başlık seçimi:

Eğer sayfaya başlık verdirilmesini istiyorsanız, **Y** seçiminden sonra ekranın altında aşağıdaki mesaj belirir:

Enter/change heading below (Esc to return):
[]

Buraya sayfaya verilmesi istenen başlık girilir. Başlık tanımlamasından sonra tekrar aşağıdaki mesaj çıkar:

Do you want a heading printed on each page ?

Her sayfaya verilecekse **Y** tuşuna basılarak, başlığın bütün sayfalara verilmesi sağlanır.

Dipnot seçimi:

Dipnot vermek için seçim yapılmışsa ekranın altında aşağıdaki mesaj çıkar:

Enter/change footing below (Esc to return):
[]

Buraya girilmesi istenen dipnot en çok **32** karakter uzunluğunda girilir.

Do you want a footing printed on each page ?

Şeklinde mesaj çıkararak, dipnotun bütün sayfalara yazdırılıp yazdırılmayacağı sorulur.

Sayfa arasında bekleme yapma:

Sayfa arasında bekleme yapma seçeneği girildiğinde aşağıdaki mesaj çıkar:

If you are printing single sheets, you will want to stop to insert each sheet.

Son sayfadan sonra sayfa başı yapma:

Bu seçim onaylandığında aşağıdaki mesaj gelir:

Do you want the last sheet printed to be ejected ?

Y harfi ile onaylandığında - "**Printer options are correct**" - iletisi çıkar.

D seçeneği ile yazım şekli:

D seçeneğini seçteğimizde **Find** ve **wiew/Edit** işlemlerinde ekranda gördüğümüz ASCII ve HEX kodları sektör sektör yazıcıdan alınacaktır.

20. DOSYA ÖZELLİKLERİNİ YAZICIYA YAZDIRMA

List

Aktif durumda bulunan sürücü ve dizindeki dosyaların adlarını, uzantılarını, kapladıkları alanı, kayıt tarih ve saatini, dosya niteliğini yazıcıya yazdırmak için **List** seçeneği kullanılır.

Bu seçeneği kullanabilmek için aşağıdaki işlemler yapılır:

L harfine basılır. Yazıcı hazır değilse ekrana aşağıdaki uyarı gelir.

```
Write fault
ESC to return, "R" to retry or "I" to ignore
```

Hazırsa aşağıdaki şekilde yazıcıdan çıktı alınır.

```
PC Tools Deluxe R4.21                               Vol Label=SISTEM DISK
Path=A:\*.*
Name      Ext      Size #Clu   Date      Time      Attributes
IO        SYS      40566    80  9/30/93   6:20a    Hidden, System, Read-Only, Archive
MSDOS    SYS      38138    75  9/30/93   6:20a    Hidden, System, Read-Only, Archive
COMMAND  COM      54619   107  9/30/93   6:20a    Normal, Archive
DBLSPACE BIN      64246   126  9/30/93   6:20a    Hidden, System, Read-Only, Archive
AUTOEXEC BAT         494     1  7/07/94  11:18p    Normal, Archive
CONFIG   SYS         284     1  6/01/94   8:24p    Normal, Archive
WINA20    386      9349    19  8/11/93   6:20a    Normal, Archive
FKLAUYE  COM      2486     5  1/01/80  12:02a    Normal, Archive
XCOPY    EXE     16930    34  9/30/93   6:20a    Normal, Archive
MERTUP   COM     20713    41  12/13/93  10:23p    Normal, Archive
DISKCOPY COM     13335    27  9/30/93   6:20a    Normal, Archive
FORMAT   COM     22916    45  9/30/93   6:20a    Normal, Archive
CHKDSK   EXE     12241    24  9/30/93   6:20a    Normal, Archive

14 files LISTed = 305707 bytes.  14 files in sub-dir = 305707 bytes.
0 files SELECTed = 0 bytes.  Available on volume = 367104 bytes.
```

21. DOSYALARI SIRALAMA

Sort

Aktif durumda bulunan sürücü ve dizindeki dosyaları istenilen niteliğe göre sıralamak için **Sort** seçeneği kullanılır.

Bu işlemi yapabilmek için aşağıdaki sıra takip edilir:

1. Belirlenen sürücü ve dizinde iken S tuşuna basılır.
2. Bu seçimden sonra ekranın altında aşağıdaki pencere çıkar:

```
Choose the sorting method desired. If it appears as desired,
enter "U" to update the directory on the disk.
F7 = by NAME F8 = by EXT F9 = by SIZE F10 = by DATE/TIME ESC = return
```

Burada geçen kullanımlar görevleri şunlardır:

U	Değişiklikleri güncelleştirerek diske kayıt etmek için.
F7=by NAME	Dosya adına göre sıralama yapar.
F8=by EXT	Dosya uzantısına göre sıralama yapar.
F9=by SIZE	Dosyaların kapladıkları alanlara göre sıralama yapar.
F10=by DATE/TIME	Dosyaların kayıt tarihlerine göre sıralama yapar.
ESC=return	Geri dönüş için.

Yukarıda belirtilen sıralama seçimlerinden birini seçtiğimizde ekrana aşağıdaki mesaj gelir:

```
A=Ascending or D=Destending [A]
Press ESC to exit
```

A=Ascending	Sıralamayı küçükten büyüğe doğru yapar.
D=Destending	Sıralamayı büyükten küçüğe doğru (tersten) yapar.

Buradaki seçeneklerden birini seçtikten sonra bir önceki pencere yeniden gelir. Burada **U** tuşuna basarsak yaptığımız değişiklik güncelleştirilerek diskte kayıt edilecektir. **Esc** tuşuna basarsak güncelleştirme yapılmadan çıkılacaktır.

22. YARDIM ALMA

Help

PCTOOLS programında çalışırken herhangi bir konuda yardım almak için **Help** seçeneği kullanılır.

Yardım almak için H tuşuna basarsız ve aşağıdaki ekran gelir:

```
PC Tools Deluxe R4.21                               Vol Label=SISTEM DISK
                                           File Functions                               Scroll Lock OFF
Path=A:\*. *

Help for "File Functions" (see the manual for detailed explanations)
Press: "C" to COPY a file or files.
      "M" to MOVE a file or files (deletes the SOURCE file).
      "O" to COMPARE the contents of files.
      "F" to FIND a character string in a file or files.
      "R" to RENAME a file or files to another name.
      "D" to DELETE a file or files.
      "U" to VERIFY a file is readable or attempt repair of a file or files.
      "E" to VIEW or EDIT (change the contents) of a file or files.
      "A" to view or change a file or files ATTRIBUTES and date/time stamp.
      "P" to PRINT the contents of a file or files.
      "W" to use the word processor on a file.
<press "Esc" to return or any other key to see more HELP information>

14 files LISTed = 305707 bytes.   14 files in sub-dir = 305707 bytes.
 0 files SELECTed = 0 bytes.     Available on volume = 367104 bytes.

Copy Move cOmp Find Rename Delete Ver view/Edit Attrib Wordp Print List
Sort Help ←=SELECT F1=UNselect F2=alt dir lst F3=other menu Esc=exit PC Tools
F8=directory LIST argument F9=file SELECTION argument F10=chg drive/path
```

Bu ekranda hangi konu ile ilgili yardım alacaksak, o konu ile ilgili tuşa basarsız ve bu konuda yardım gelir. Paga Up/Page Down tuşlarına basarak önceki/sonraki sayfayı görebiliriz. Burada da Page Down tuşuna basınca ikinci bir sayfa gelecektir.

Yardım menüsünde bulunan seçeneklerin anlamları:

- "C" Kopyalama konusunda açıklayıcı bilgiler.
- "M" Dosya taşıma konusunda yardım.
- "O" Dosya karşılaştırma konusunda yardım.
- "F" Karakter grubu arama konusunda yardım.
- "R" Dosya adı değiştirme konusunda yardım.
- "D" Dosya silme konusunda yardım.
- "E" Dosya içeriğini görme konusunda yardım.
- "A" Dosya niteliklerini değiştirme konusunda yardım.
- "P" Yazıcıya yazdırma konusunda yardım.
- "W" Kelime işlem bölümü konusunda yardım.
- "L" Dosya özelliklerini yazıcıdan alma konusunda yardım.
- "S" Dosyaları sıralama konusunda yardım.
- "F1" İşaretlenen dosyalardan işaretlemeleri kaldırma konusunda yardım.

- “F2” Dosya niteliklerin daha açık görme konusunda yardım.
“F3” Disk işlemleri menüsüne geçiş konusunda yardım.
“F8” * ve ? jokerleri ile istenilen özellikte dosyaları görüntüleme yardım.
“F9” * ve ? jokerleri ile istenilen özellikte dosyaları işaretlemeye yardım.
“F10” Sürücü ve dizin değiştirme konusunda yardım.
“ESC” Çıkış ve geri dönüş için.

F. DİSK İŞLEMLERİ MENÜSÜNÜN AMACI ve KOMUTLARI

Şimdiye kadar gördüğümüz bütün işlemler dosya işlemleri ile ilgili idi. PCTOOLS'un dosya işlemleri yanında disk işlemlerini de yapabilme becerisini de **Disk and Special Functions** bölümünde göreceğiz.

Bu bölümde disk ile ilgili hangi işlemleri, nasıl yapabileceğimizi göreceğiz. Bunlar disket kopyalama, disket karşılaştırma, disket formatlama, disket haritası gibi bölümlerdir.

Disk işlemleri menüsüne geçmek için, PCTOOLS'un açılış menüsünde iken **F3** tuşuna basarak geçebileceğimiz gibi, dosya işlemleri menüsünde iken de **F3** tuşuna basarak Disk İşlemleri Menüsü'ne geçebiliriz.

F3 tuşuna bastığımızda Disk İşlemleri Menü'sü aşağıdaki biçimde karşımıza gelir:

```
PC Tools Deluxe R4.21      Disk and Special Functions
-----
DISK SERVICES: Copy cOmpare Find Rename Verify view/Edit Map Locate iNitalize
SPECIAL SERVICES: Directory maint Undelete system Info Park Help
F3=file srvc F10=F3+chg drv Esc=exit PC Tools
```

Bu ekranda disk işlemleri ile ilgili seçimlerimizi yapabiliriz.

G. DİSK SERVİS KOMUTLARI

1. DOSYA İŞLEMLERİ MENÜSÜNE GEÇME

F3=file srvc

Disk işlemleri menüsünde iken dosya işlemleri menüsüne geçmek için **F3** tuşuna basılır. F3 tuşunun genel görevi burada menü geçişini sağlamaktır.

2. SÜRÜCÜ DEĞİŞTİRME

F10=F3+chg drive

Sürücü değiştirmek için **F10** tuşuna basılır. Sürücü değiştirilince, yeni sürücü dosya işlemleri menüsü ile karşımıza gelir. Disk işlemleri menüsünde bütün işlemler için hangi sürücüde yapılacağı sorulmaktadır.

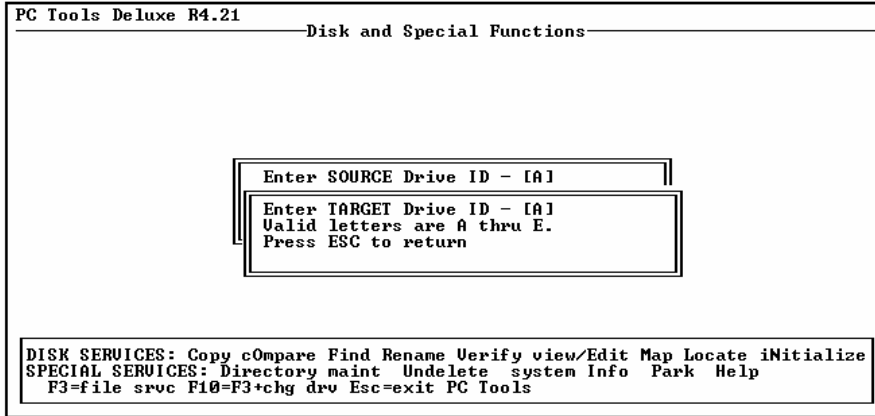
3. DİSKET KOPYALAMA

Copy

Bir disketi başka bir diskete aynen kopyalamak gerektiğinde PCTOOLS ile de bu işlemi yapabiliriz. Burada yapacağımız işlem MS-DOS'daki DISKCOPY komutunun karşılığıdır.

Disket kopyalamak için aşağıdaki işlemler takip edilir:

1. Kaynak disket A sürücüsüne takılır.
2. Disk işlemleri menüsünde iken C tuşuna basılır ve aşağıdaki mesaj çıkar:



Yazma sırasında **W** ve **F** harfleri görünecektir. **F=Formatting** (formatlıyorum), **W=Writing** (yazıyorum) anlamındadır. **Nokta** (.) işaretleri yazmanın yapıldığını bildirir. Eğer **E** harfi çıkarsa, disketin o sektörünün bozuk olduğunu belirtir.

Disketin belli bir bölümü okunmuş ve yazılmıştır. Daha kopyalama bitmemiştir. Kaynak disketi ve hedef disketi yeniden isteyecektir. Bu işlemi kopyalama bitene kadar sürdüreceğiz.

Kopyalama işlemi tamamlandıktan sonra:

Diskcopy completed.

mesajı ile kopyalamanın tamamlandığı bildirilecektir.

Yazma işlemine başlamadan önce disketiniz **protect** ise ekranın altında, aşağıdaki pencere açılacaktır:

**Write attempt on protected diskette
Press ESC to return, "R" to retry**

Böyle bir durumda disketinizi çıkartın ve **protect**'ini açarak yeniden takın ve **R** tuşuna basın. İşlem devam edecektir.

Disketinize yazma sırasında, bozuk ve yazma yapılamıyorsa bu kez de aşağıdaki pencere açılır:

**Bad CRC on diskette read
Press ESC to return, "R" to retry**

Böyle bir durumda disketinizi değiştirmelisiniz.

4. DİSKET KARŞILAŞTIRMA

cOmpare

İki disketi karşılaştırmak istediğimizde **cOmpare** komutunu seçeriz. Disket karşılaştırmak için aşağıdaki işlemler sırası ile yapılır:

1. Disk işlemleri menüsünden **O** tuşuna basılır.

2. Disket kopyalamada olduğu gibi bir pencere açılarak kaynak disketin hangi sürücüye takılacağı sorulur. Sürücü belirtildikten sonra bu kez, hedef disketin hangi sürücüye takılacağı sorulur. Burada da sürücü belirtilir. Ekran görüntüsü aşağıdaki gibi üstüste iki pencere şeklinde oluşur.

```
Enter FIRST Drive ID - [A]
Enter SECOND Drive ID - [A]
Valid letters are A thru E.
Press ESC to return
```

FIRST=Orijinal, kaynak disket. **SECOND**=Karşılaştırılması yapılacak, hedef disket.

A sürücüsüne birinci disketin takılması için aşağıdaki mesaj çıkar:

```
Insert First diskette in drive A
Press any key to continue
```

Disketi taktıktan sonra **Enter** tuşuna basarız. Disket kopyalamada olduğu gibi bir pencere gelerek disketi okumaya başlayacak ve okuma sırasında;

```
Comparing diskette in drive A to diskette in drive A
```

mesajı ekranda görünecektir.

Okuma bitince;

```
Insert Second diskette in drive A
Press any key to continue
```

mesajı ile ikinci disketi takmamızı isteyecektir. İkinci disketi takarak **Enter** tuşuna bastığımızda karşılaştırma başlayacaktır.

Karşılaştırma sırasında **C** harfi karşılaştırmanın yapılmakta olduğunu, **nokta** (.) ise karşılaştırmanın normal olduğunu gösterir.

Karşılaştırma sonunda;

```
Diskette Compare completed
Press any key to continue
```

disket karşılaştırmasının tamamlandığını ve bir tuşa basmamızı ister.

Karşılaştırma sırasında bir hata varsa, disketinizdeki hatanın durumuna göre aşağıdaki gibi bir mesaj çıkar:

```
Disk MISMATCH in logical sector num 0000000 at displacement 067
Mismatched byte : " " (BC) (in 1st disk) " " (33) (in COMPARE TO disk)
Press any key to continue
```

Bu mesaj iki disketin farklı olduğunu bildirmektedir.

Disket kopyalama ve disket karşılaştırmada iki diskin de aynı özellikte olması gerekir. DD ile DD, HD ile HD ve aynı boyutta olan disketler için geçerlidir.

5. DİSKTE KARAKTER GRUBU ARAMA

Find

Bir dosya ya da dosyada değilde, disketin tümünde bir karakter grubu aranmak istendiğinde **Find** seçeneği kullanılır.

Bu seçeneği kullanmak için aşağıdaki sıra takip edilir:

1. Disk işlemleri menüsünden F tuşuna basılır.
2. Aranılacak disket sürücüsü tanımlanır. Bunun için ekrana şöyle bir pencere açılır:

```
Enter Drive ID - [A]
Valid letters are A thru E.
Press ESC to return
```

Bundan sonra yapılacak işlemler dosya işlemlerinde yaptığımız arama biçiminin aynıdır.

6. DİSK ETİKETİ DEĞİŞTİRME

Rename

Diske verilen etiket adı (Volume) değiştirmek istendiğinde **Rename** seçeneği kullanılır.

Bu işlem için aşağıdaki sıra takip edilir:

1. Disk işlemleri menüsünden R tuşuna basılır.
2. Hangi sürücüde değişiklik yapılacağı tanımlaması için aşağıdaki pencere açılır:

```
Enter Drive ID - [A]
Valid letters are A thru E.
Press ESC to return
```

Belirlenen sürücüye disket takılır ve **Enter** tuşuna basılır. Bu kez aşağıdaki ekran gelir:

```
Drive A
Current volume label=SISTEM DISK
Enter the new volume label [SISTEM DISK]
Press ESC to Exit
```

Disk etiketimizin adı örneğimizde SISTEM DISK idi. **Enter the new volume label**'ın karşısına yeni ad yazılarak **Enter** tuşuna basılır.

```
Volume renamed successful.
Press any key to continue
```

Mesajıyla değişikliğin yapıldığı onaylanır.

7. SEKTÖR TARAMA

Verify

Diskteki sektörleri okutup bozuk alanları belirlemek gerektiğinde **Verify** komutu kullanılır.

İşlem için aşağıdaki sıra takip edilir:

1. Disk işlemleri menüsünden V tuşuna basılır.
2. Hangi sürücünün kontrol edileceğini belirten pencere açılır. Bu pencereye kontrol edilecek sürücü harfi girilir. Aşağıdaki mesaj çıkar:

```
Drive A is about to be verified,
Press any key to continue
Press ESC to Exit
```

Bu mesaj A sürücüsündeki disketin kontrolü yapılacağını, başlamak için herhangi bir tuşa geri dönmek için **Esc** tuşuna basılacağını belirtir. Bundan sonra kontrol işlemi başlayacak ve aşağıdakine benzer bir mesaj çıkacaktır:

```
Drive A being verified.
Reading logical 0000234 thru 0000251
Press ESC to Exit
```


Ekranın tuş kullanımı:

Home=beg of file/disk	Diskteki ilk sektöre gider.
End=end of file/disk	Diskteki son sektöre gider.
Esc=Exit	Çıkış için.
PgDn=forward	Yarım sektör ileri gider.
PgUp=back	Yarım sektör geri gider.
F2=chg sector num	Diskette belirlenen sektöre geçmeyi sağlar.
F3=Edit	Görülen sektörde değişiklik yapılmasını sağlar.
F4=get name	Görüntülenen sektörün dosya adının görüntülenmesini sağlar.

F2 tuşuna basıldığında ekrana aşağıdaki gibi bir seçenekler penceresi gelir:

```
Indicate which sector to display;
"B" = BOOT sector
"F" = First FAT sector
"R" = First ROOT directory sector
"D" = First DATA sector
"C" = Enter a cluster # (2-02848)
nnnnn = sector number
```

Buradaki tuşların görevleri aşağıdaki gibidir:

"B"	BOOT sektörü görüntüler.
"F"	FAT sektörünün ilk bilgilerini verir.
"R"	ROOT sektörün ilk bilgilerini verir.
"D"	DATA sektörüne ait ilk bilgileri verir.
"C"	Erişilmesi istenen blok numarası girilerek, istenen bloğu görüntüler.
"nnnnn"	Sektör numaralarının girilebilecek en büyük boyutu.

Belirlenen sektörde **Edit** işlemi, yani değişiklik yapılacaksa, dosya işlemlerinde gördüğümüz kurallar burada da geçerlidir.

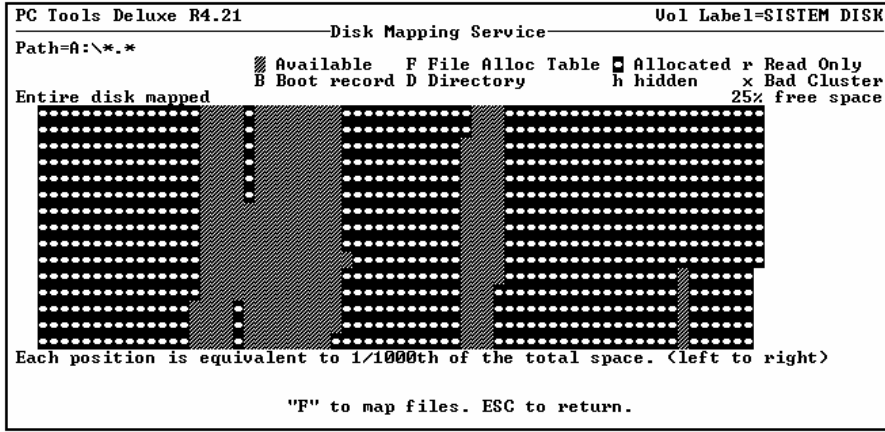
9. DİSKİN YAPISAL DURUMUNU GÖRME

Map

Bir diskin yapısal durumunu kontrol etmek ya da görmek için **Map** komutu kullanılır. Burada diskte bulunan bozuk sektörleri, dosyaların kapladıkları alanları, diskte kalan boş yeri rahatlıkla görebiliriz. Kısaca disketin haritasını görebiliriz.

Diskin yapısal durumunu görmek için aşağıdaki işlemler yapılır:

1. Disk işlemleri menüsünde M tuşuna basılır.
2. Hangi sürücünün yapısal durumu görülecekse, çıkan pencereye o sürücünün harfi girilir. Bundan sonra aşağıdaki ekran gelir:



Burada gördüğünüz işaret ve terimlerin anlamları:

Available	Diskte kalan boşyerler.
B Boot record	Boot sektöre ait olan yerler. Kontrol dosyaları.
F File Alloc Table	FAT alanı. Dosyalara ait bilgilerin bulunduğu ortam. Dosyanın kapladığı alan, kayıt tarihi ve saati gibi...
D Directory	Dosyaların adlarının bulunduğu yerler.
Allocated	Diskte dosyalar tarafından dolu olan yerler.
h hidden	İşletim sistemine ait gizli dosyaların bulunduğu yerler.
r Read Only	İşletim sistemine ait yalnız okunabilir dosyaların bulunduğu yerler.
x Bad Cluster	Diskte bulunan bozuk sektörler. Bu işaret yapıp söner.
xx free space	Diskte kalan boşluk. Yüzde olarak.
Entire disk mapped	Diskteki dosyaların durumları. (Haritası)
"F" to map files	Dosyaların diskteki alanlarını izlemek için F tuşuna basılması gerektiği.

F tuşuna basarak dosyaların diskin hangi konumunda olduğunu tek tek görebiliriz. F tuşuna basınca ekranın altında aşağıdaki mesaj çıkacaktır:

```
Select 1st file to be mapped and then press "G" to proceed.  
F8=directory LIST argument F9=file SELECTION argument F10=chg path  
↑↓=scroll ←=SELECT F1=UNselect F2=alt dir 1st Esc=exit
```

Buradaki tuş kullanımlarının anlamları şunlardır:

"G"	Bu tuş işleme başlamak içindir.
F8=directory LIST argument	İstenen özellikte dosyaları ekranda görmek için.
F9=file SELECTION argument	İstenen özellikte dosyaları işaretlemek için.
F10=chg path	Sürücü değiştirmek için.
↑↓=scroll	Işıklı göstergelyi aşağı yukarı hareket ettirmek için.
←=SELECT	Dosya işaretlemek için.
F1=UNselect	İşaretlenen dosyalardan işaretleri kaldırmak için.
F2=alt dir 1st	Dosyaların özelliklerini detayları ile göstermek için.
Esc=exit	Çıkış için.

Dosyalar F8 ya da F9'la seçilmişse ekrana dosyanın konumu geldiğinde aşağıdaki mesaj gelir:

```
Use arrows(← →) to view other files.  
"D" to view entire disk map. "F" to reselect files. ESC to return.
```

Burada sol/sağ ok tuşları ile dosyaları tek tek görebiliriz. F tuşuna basınca dosya konumuna, D tuşuna basınca diskin genel gösterimine geçer.

10. DOSYALARIN ÇALIŞMA DİZİNLERİNİ GÖRME

Locate

Dosyaların hangi dizinde bulunduğunu görmek ve dosya hakkında bilgi almak için **Locate** komutu kullanılır.

Bu işlemi yapabilmek için aşağıdaki sıra takip edilir:

1. Disk işlemleri menüsünde L tuşuna basılır.
2. Hangi sürücüde arama yapılacağını soran pencereye, sürücü harfi girilir. Bundan sonra hangi özellikte programların görüntüleneceğini soran aşağıdaki ekran gelir:

```
PC Tools Deluxe R4.21          File Locator Service
Path=A:

Enter the name and/or extension to be used as a search
argument for the directory listing. A question mark (?)
is a "WILDCARD" which will always match any character
found in that position. An asterisk in the name or
extension will match that position and all remaining
positions.

Name=[*      ]
Ext =[*     ]

Name and Extension correct as entered, <Y/N)? [y]

Press ESC to Exit
```

Bu ekranda görüntülenmesini istediğimiz dosyaların özelliklerini girer ve **Enter** tuşuna basarız. Görüntü aşağıdaki gibi olacaktır.

```
PC Tools Deluxe R4.21          File Locator Service          Vol Label=SISTEM DISK
Path=A:\MASA\WINDOWS
Searching Path = A:\*. *
IO.SYS          40566    9/30/93    6:20a
MSDOS.SYS      38138    9/30/93    6:20a
COMMAND.COM    54619    9/30/93    6:20a
AUTOEXEC.BAT   494      7/07/94    11:18p
CONFIG.SYS     284      6/01/94    8:24p
Searching Path = A:\DOS\*. *
ANSI.SYS       9065     9/30/93    6:20a
DELTREE.EXE    11111    9/30/93    6:20a
DISKCOMP.COM   10748    9/30/93    6:20a
Searching Path = A:\PW\*. *
PW.COM         4051     2/04/88    12:09a
PW.NET         60       4/20/93    1:10p
PFS_DIR        236     1/01/80    12:28a
Searching Path = A:\PW\OZEL\ISYAZI\*. *
DERSPROG.DOC   7529    12/09/93    8:36p
KARAKTER.DOC   3871    12/15/93    9:55p
Searching Path = A:\WINDOWS\WINWORD\*. *
SOZLUK.DOC     16971   11/16/92    2:39p
ARKAKAP.DOC    2606    11/25/92    7:50p

Press any key to continue or ESC to exit.
```

11. DİSKET FORMATLAMA

iNitalize

Disket formatlamak istediğimiz zaman **iNitalize** komutunu kullanırız.

Disket formatlamak için aşağıdaki sıra takip edilir:

1. Disk işlemleri menüsünde N tuşuna basarız.
2. Hangi sürücüde disketin formatlanacağı sorulan pencereye, sürücü harfi girildikten sonra Enter tuşuna basarız. Aşağıdaki ekran gelir:

```
PC Tools Deluxe R4.21 —Disk Initialization Service—

Drive A is about to be initialized <formatted>.

Choose the formatting desired
and press ENTER to begin.

720K
144M ← 3 1/2 Diskette, 80 tracks,
      Double-sided, 18 sectors per track

Press ESC to Exit
```

Burada disketin hangi kapasitede formatlanacağı sorulmaktadır. İstenen kapasite seçilir ve Enter tuşuna basılır. Aşağıdaki mesajla formatlamanın başlayacağı bildirilir:

```
Drive A is about to be initalized ( formatted ).
Press any key to continue
```

Aşağıdaki pencere gelerek formatlama olayı başlar:

```
PC Tools Deluxe R4.21 —Disk Initialization Service—

Drive A is being initialized <formatted>.
Double-sided, 18 sectors per track

1 2 3 4 5 6 7
012345678901234567890123456789012345678901234567890123456789
.....EEE.....F
.....

Press ESC to Exit
```

Formatlama sırasında F işareti formatlamanın yapılmakta olduğunu, V kontrol edildiğini belirtir. **Nokta** (.) işareti normal olduğunu, **E** işareti ise o sektörün bozuk olduğunu gösterir.

Formatlama bitiminde etiket adının girilmesi için aşağıdaki mesaj gelir:

```
Enter the new volume label [          ]
```

Disk etiket adı girildikten sonra **Reserving space for system files...** mesajı belirir ve aşağıdaki ekran gelir:

```
PC Tools Deluxe R4.21      Disk Initialization Service
-----

Would you like this to be a "bootable" <system> disk <Y/N>?

If you answer "Y" <YES>, you must also, from a DOS
prompt, execute the SYS command and then, after
returning to PC Tools, copy COMMAND.COM to the new
diskette from your DOS system diskette. If you are
NOT running IBM PC-DOS, you must first delete the
IBMBIO.COM and IBMDOS.COM files with PC Tools. This
option does reduce the useable space on the new diskette.

Press ESC to Exit
```

Burada disketin sistem disketi yapılıp yapılmayacağı sorulmaktadır. Eğer sistem disketi yapacaksak **Y**, yapmayacaksak **N** harfi girilir. Sistem disketi yapacaksak PCTOOLS'dan çıkar ve **SYS.COM** ile sistem transferi yaparak, ardından **COMMAND.COM** dosyasını kopyalarız.

Sonuçta aşağıdaki mesajla disket hakkında bilgi verilir.

```
1457664 bytes total disk space.
  41984 bytes in hidden/system files.
    0 bytes in bad sectors.
1415680 bytes available on disk.
```

```
Format complete.
Press any key to continue.
```

Herhangi bir tuşa bastığımızda formatlama olayı bitmiş olur.

12. DİZİN İŞLEMLERİ

Directory maint

Dizin ile ilgili işlemler yapmak istediğimizde **Directory maint** komutu kullanılır.

Dizin işlemlerini yapmak için aşağıdaki sıra takip edilir:

1. Disk işlemleri menüsünde iken **D** tuşuna basılır.
2. Hangi sürücüde işlem yapılacağını soran pencereye sürücü harfi girildikten sonra aşağıdaki ekranlar gelir:

Eğer disketinizde daha önce hiç dizin açılmamışsa aşağıdaki mesaj:

**The current volume contains no sub-directories.
If yo wish to create a sub-directory, press "Y",
else press "Esc" to return to menu.**

Disketinizde daha önce dizin açılmışsa aşağıdaki ekran gelir:

```
PC Tools Deluxe R4.21          Uol Label=SISTEM DISK
Directory Maintenance Service  BLINK=DOS current
Path=A:\
R
O
O
T
- DOS
  - PW
    - DATA
    - OZEL
      - ISYAZI
        - KAMU
        - SIRKET
      - KISISEL
    - KITAP
    - PREMS
    - TETRIS
  - DILLER
    - BASIC
    - COBOL
    - PASCAL
  - WINDOWS
    - SYSTEM
    - CORELDRW
    - WINWORD

Use cursor control keys to follow the chain to the desired directory.
Then choose a directory maintenance option below, or Esc to exit.
F1=rename F2=create F3=remove F4=chg DOS current directory F5=prune & graft
```

Bundan sonra hangi dizin içerisinde çalışılacaksa yön tuşları ile istenen dizinin üzerine gelinir ve hangi işlemi yapmak istiyorsak, o işlem tuşuna basılır.

Bu ekrandaki tuş kullanımlarının görevleri aşağıdaki gibidir:

F1=rename	Dizin adı değiştirme.
F2=create	Yeni dizin açma.
F3=remove	Dizin silme.
F4=chg DOS current directory	DOS dizininden çıkarak başka bir dizini aktif duruma getirme.
F5=prune & graft	Bir dizini başka bir dizine taşıma. Ekleme.

Dizin adı değiştirme: F1=Rename

Dizin adı değiştirmek için bir dizini işaretler F1 tuşuna basarız. Ekranın altında eski dizin adı belirir. Buraya yeni dizin adını girerek Enter tuşuna basarız.

Dizin açma: F2=create

Dizin açmak için hangi dizinin altına yeni bir izin açılacaksa o dizinin üzerine geliriz ve F2 tuşuna basarız. Ekranın altında yeni izin adını girebilmemiz için mesaj gelir. Dizin adını girer ve Enter tuşuna basarız.

Dizin silme: F3=remove

Sililmek istenen dizinin üzerine gelinir ve F3 tuşuna basılır. Silmek isteyip istemediğimiz sorulduğunda silme onaylanarak izin silinir. Bir dizini silmek için dizinin içerisinde dosya olmaması gerekir. Böyle bir durumda aşağıdaki mesaj gelir:

**The directory not empty
Press any key to continue**

DOS dizininin yetkisini aktarma: F4=chg DOS current directory

ROOT dizininin yetkisi geçici olarak başka bir dizine verilmek istendiğinde F4 tuşuna basarız. İşlem sonunda aktardığımız dizinin yetkisini yine F4 tuşu ile iptal etmemiz gerekir.

Dizin taşıma-ekleme: F5=prune & graft

Dizin taşıma ya da ekleme için istenen izin üzerine gelinir. F5 tuşuna basılır ve taşıma komutu verilmiş olur. Taşınacağı yere getirilir, F5 tuşuna yine basılır.

13. SİLİNİMİŞ DOSYA VE DİZİNİ KURTARMA

Undelete

Daha önce silinmiş bir izin ya da dosyayı kurtarmak için **Undelete** komutu kullanılır. Bildiğimiz gibi DOS izin ve dosya silme işleminde içeriklerini değil yalnızca adının ilk karakterini siliyordu. Burada dikkat edeceğimiz nokta, silinen izin ya da dosyanın üzerine herhangi bir kayıt yapılmış olmaması gerekiyor.

Silinen bir dosya ya da dizini kurtarmak için sırasıyla aşağıdaki işlemler yapılır.

1. Disk işlemleri menüsünde iken U tuşuna basılır.
2. Kurtarmanın hangi sürücüde yapılacağı sorusuna istenen sürücü harfi girilir. Ekranı aşağıdaki pencere gelir.

```
Please choose FILE or SUB-DIRECTORY:
File ← → Undelete a file
Sub-Dir
```

Bu pencerede dosya mı yoksa dizin mi kurtarılacağı sorulmaktadır. Dosya ise **File**, dizin ise **Sub-Dir** seçeneği seçilir. Seçeneklere geçiş aşağı/yukarı ok tuşu ile yapılır.

Dosya seçimini örnek alacak olursak, seçim sonrası Enter tuşuna basıldığında disk taranarak silinen dosyalar ilk karakterleri ? işaretli olarak listelenecektir. Silinmiş herhangi bir dosya yoksa **No entries found** mesajı çıkar. Silinmiş dosya örneği aşağıdaki gibi olur:

```
PC Tools Deluxe R4.21                               Vol Label=SISTEM DISK
Undelete Service                                     Scroll Lock OFF
Path=A:\*. *
Name Ext Size Attr Date Name Ext Size Attr Date
?UTOEXEC BAT 494 ... 7/07/94
?ONFIG SYS 284 ... 6/01/94
?M2 EXE 82328 ...A 6/03/90
?ISRCOPY COM 13335 ... 9/30/93
?C02 GIF 5935 ...A 7/04/94
?ORMAT COM 22916 ... 9/30/93
?MBASIC EXE 78864 ...A 7/07/86

@ = Automatic recovery possible
Select file(s) to be un-deleted and then press "G" to proceed.
F8=directory LIST argument F9=file SELECTION argument F10=chg drive/path
↑=scroll ←=SELECT F1=UNselect F2=alt dir lst Esc=exit
```

Ekranda gördüğünüz gibi daha önce silinen dosyaların listesi geldi. Kurtarma işlemleri için alt kısımda görünen terim ve tuş kullanımlarının görevleri aşağıdadır:

- @ Otomatik olarak kurtarılabilecek dosyalar.
- "G" Kurtarmayı başlatmak için basılacak tuş.

Diğer tuşların kullanımları daha önceki görevleri ile aynıdır.

Bundan sonra kurtarılacak dosya ya da dosyalar işaretlenerek **G** tuşuna basılır ve aşağıdaki mesaj gelir:

```
PC Tools Deluxe R4.21                               Vol Label=SISTEM DISK
Undelete Service                                     Scroll Lock OFF
Path=A:\
Name Ext Size #Clu Date Time Attributes
?UTOEXEC BAT 494 1 7/07/94 11:18p Normal

Enter first character-?UTOEXEC.BAT
```

Burada kurtarılabak dosyanın ilk karakteri ? işaretli olarak gelir. Bizden istenen dosyanın ilk karakterinin girilmesidir. Dosyanın ilk karakterini unutmuşsanız, çalışmalarınızı etkilemeyecek bir dosya ise herhangi bir harf girebilirsiniz. Bundan sonra aşağıdaki ekran gelecektir:

```
PC Tools Deluxe R4.21          Undelete Service          Vol Label=SYSTEM DISK
                               Scroll Lock OFF
Path=A:\
Name      Ext      Size #Clu   Date      Time  Attributes
AUTOEXEC  BAT      494      1   7/07/94  11:18p  Normal
-----
                               Use function keys to make selection
                               F1 - Automatic selection of clusters
                               F2 - Manual selection of clusters
                               ESC - Return
```

Bu ekranda iki türlü kurtarma yöntemi vardır:

- F1** Otomatik kurtarma.
- F2** Görerek sektör sektör kurtarma.

F1 tuşuna basınca kurtarma otomatik olarak gerçekleştirilecektir. Ekranda aşağıdaki mesaj ile kurtarmanın hatasız gerçekleştirildiği bildirilecektir:

File was successfully Undeleted - Press any key to continue

Bundan sonra herhangi bir tuşa basarak işlem tamamlanır. Birkaç dosya işaretlemişsek bir tuşa basınca bundan sonraki kurtarılabak dosya gelecek aynı işlemleri yeniden yapacağız.

Sektör sektör kontrol ederek kurtarma yapmak istendiğinde F2 tuşu ile kurtarma yapılır. Dizin kurtarmak istendiğinde aynı işlemler yapılacaktır.

14. SİSTEM BİLGİSİ

Info

Bilgisayar ve yan birimleri hakkında bilgi edinmek istendiğinde **Info** komutu kullanılır. Bu işlem için disk işlemleri menüsündeyken I tuşuna basılır ve aşağıdaki ekrana benzer biçimde sistem bilgileri gelir:

```
PC Tools Deluxe R4.21      System Information Service
      Computer - IBM/PC AT
The BIOS programs are dated - 11/11/92
      Operating system - DOS 6.20
Number of logical disk drives - 5
      Logical drive letter range - A thru E
      Serial Ports - 2
      Parallel Ports - 1
      CPU Type - 80386
Relative speed (orig PC=100%) - 2810%
      Math co-processor present - No
User programs are loaded at HEX paragraph - 09D4
Memory used by DOS and resident programs - 40256 bytes
      Memory available for user programs - 615088 bytes
      Total memory reported by DOS - 639K
PC Tools has found the total memory to be - 640K
Enhanced Graphics Adapter present (color) - 256K
      Expanded memory total - 1024K, in use - 1024K
Additional ROM BIOS found at HEX paragraph - C000K

Press any key to return
```

Burada görülen terimlerin anlamları şöyledir:

Computer	Bilgisayarın tipi, modeli.
The BIOS programs are dated	BIOS programının yerleştirildiği tarih.
Operating system	Kullanılan işletim sistemi ve versiyonu.
Number of logical disk drives	Kullanılan sürücücü sayısı.
Logical drive letter range	Kullanılabilen sürücü harfi tanımlaması.
Serial ports	Seri bağlantı sayısı.
Parallel ports	Paralel bağlantı sayısı.
CPU type	Bilgisayarın mikroişlemcisi.
Relative speed (orig PC=100%)	Ana bellek erişim hızı.
Math co-processor present	Matematik işlemci tipi.
User programs are loaded at HEX paragraph	Kullanıcı programlarının bellekteki başlangıç adresi.
Memory used by DOS and resident programs	DOS ve bellekte kalıcı programların o an bellekte kapladığı alan.
Memory available for user programs	Bellekte kalan boş alan.
Total memory reported by DOS	DOS'un saptadığı toplam bellek.
PC Tools has found the total memory to be	PC Tools'un saptadığı toplam bellek.
Enhanced Graphics Adapter present (color)	Kullanılan ekran tipi.

15. HARD DİSKİ PARK ETME

Park

Eski tip hard disklerin kafası kapanma sonrasında oynar durumda kalıyordu. Bu da bilgisayarı yerinden oynatınca hard diskin zarar görmesine neden oluyordu. Bu nedenle bilgisayarı kapatırken park ederek kafa yerine oturtuluyordu. Yeni tip hard disklerde bilgisayarı kapatınca otomatik olarak park yapıldığından böyle bir sorun yoktur.

Park yapabilmek için, disk işlemleri menüsünde iken P tuşuna basmak gerekir. Bundan sonra hard diskin park edildiği mesajı verilecektir.

16. YARDIM ALMA

Help

Disk işlemleri menüsünde iken, bu bölümle ilgili yardım alabilmek için H tuşuna basmamız gerekir. Bundan sonra aşağıdaki ekran gelecektir:

```
PC Tools Deluxe R4.21      Disk and Special Functions
-----
Help for "Disk and Special Functions" (see manual for detailed explanations)
Press: "C" to COPY a removable disk to another removable disk.
       "O" to COMPARE the contents two removable disks.
       "F" to FIND a character string on a disk.
       "R" to RENAME a disk volume label.
       "U" to VERIFY a disk is readable or attempt repair of a disk.
       "E" to VIEW or EDIT (change the contents) of a disk.
       "M" to MAP the contents of a disk and where files reside on a disk.
       "L" to LOCATE in which directories a file or files reside.
       "N" to INITIALIZE (FORMAT) a removable disk.
       "P" to park hard disk heads for system relocation.
<press "Esc" to return or any other key to see more HELP information>

DISK SERVICES: Copy cOmpare Find Rename Verify view/Edit Map Locate iNitialize
SPECIAL SERVICES: Directory maint Undelete system Info Park Help
F3=file srvc F10=F3+chg drw Esc=exit PC Tools
```

Yardım almak için ekranda görülen tuşlara bastığımızda o konu ile ilgili yardım alabiliriz. Disk işlemleri menüsünde iken kullanılan tuşa bu ekranda basınca, belirtilen konuda yardım alınır. Pg Dn tuşuna basarak bir sonraki yardım sayfasını, Pg Up tuşuna basarak bir önceki yardım sayfasını alabiliriz.

PCTOOLS programı gördüğümüz gibi yetenekli bir programdır. Bütün bu öğrendiklerimizden sonra artık bu programı rahatlıkla kullanabiliriz.

SORULAR

1. Bellek hizmet programlarının özelliği nedir?
2. Dosya İşlemleri Menü'sü ile Disk İşlemleri Menü'sünün görevleri nelerdir?
3. Aynı özellikteki dosyaları hangi menüde, hangi tuşla işaretletiriz?
4. PCTOOLS'ta disket formatlama hangi seçenекle yapılır?
5. Info seçeneğinin görevi nedir?
6. F3 fonksiyon tuşunun menülerdeki genel görevi nedir?
7. UNSELECT seçeneğinin yaptığı görev nedir?
8. PARK seçeneğinin görevi nedir?
9. Disk/disketin haritasını hangi seçenекle görürüz?
10. F10 tuşunun genel görevi nedir?
11. view/Edit seçeneğinin görevi nedir?
12. Bir dizinden başka bir dizine nasıl geçeriz?
13. Bir sürücüden başka bir sürüçüye nasıl geçeriz?
14. Dizindeki dosyaları hangi seçenекle istenen biçimde sıralayabiliriz?
15. PCTOOLS ile DOS arasındaki avantaj ve dezavantajlar nedir?

BÖLÜM VI.

BELE DÜZENLEYİCİ (KELİME İŞLEM, EDİTÖR) PROGRAMI

A. GİRİŞ

Editör programları, yazışmalarda ve çeşitli belgelerin hazırlanmasında kullanıcıya büyük kolaylıklar sağlar. Bir dökümanı ekranda istediğimiz şekilde hazırladıktan sonra kağıda aynı şekil ve özelliklerde çıktı alabiliriz. İşte bu şekilde yazılarımızı ve dökümanlarımızı hazırlayabildiğimiz, herkesin kolaylıkla kullanabildiği programlara **editör** denir. Editörler, kelime işlem programları olarak da anılırlar.

B. EDİTÖR PROGRAMLAMANIN TEMEL MANTIĞINI KAVRAMA

Editör programları kağıda aktarabileceğimiz bilgilere, ekranda son şeklini verene kadar istediğimiz düzeltmeyi yapmamızı sağlar. En gelişmiş daktilodan daha becerikli ve daha kullanışlıdır. Yazılacak metni hazırlama, düzeltme, saklama, başka bir metin ile birleştirme, tekrar çağırma ve yazıcıdan çıktı almayı sağlarlar. Bir editör programı ile çalışmanın en güzel yanı, dökümanımıza son şeklini verdikten sonra kağıt çıktısını alabilmektir. Yanlışlarımızı ekranda düzeltebildiğimiz için temiz bir çıktı almamızı sağlar. Böyle bir çalışma ile hem zamandan hem de kağıttan tasarruf etmiş oluruz.

Editör programlarının hepsinin amacı ve mantığı aynıdır. Kullanım açısından ufak tefek değişiklikler gösterirler. Birbirlerine oranla kullanım kolaylığı ve yapabildiği beceriler açısından farklılıklar vardır. Editör programları sürekli gelişmektedir. Yeni yeni uyarlamalarını (version) kullanıcıya sunmaktadırlar. Bu da, bu alanda daha iyi yazılımların ortaya çıkmasını sağlamaktadır.

C. EDİTÖR PROGRAM ÇEŞİTLERİ

Kullanıcılar bir çok editör programı arasında seçim yapmakta zorlanmaktadırlar. Çünkü her birinin diğerinden farklı ve üstün bir yanı vardır.

Kullanıcılar tarafından en çok kullanılan ve beğenilen editör programlarını şöyle sayabiliriz:

PROFESSIONAL WRITE (PW)
WORDSTAR (WS)
MICROSOFT WORD
WORD PERFECT
CWRITER
WORD FOR WINDOWS gibi...

D. BELGE DÜZENLEYİCİ (EDITOR) PROGRAM KAVRAMLARI

Editör programlarının kullanımında sık sık kullanılan ve karşılaşılabileceğimiz terim ve kavramlar şunlardır:

Bloklama: Döküman içerisinde belirli bir bölümü işaretleyerek, o bölümle ilgili çalışma yapmak gerekebilir. Bunlar taşıma, silme, kopyalama ve kağıda çıktı alma gibi işlemlerdir. İşte bu işlemi yaparken bir bölümün işaretlenmesi işlemine bloklama diyoruz.

Edit: Düzeltme yapmak anlamında kullanılır. Bir dökümanı yazmak, geliştirmek, gerekli düzeltme işlemlerini yapmaktır.

Cut: Kesmek anlamındadır. Herhangi bir bölümü blokladıktan sonra kesmek/silmektir.

Paste: Yapıştır anlamındadır. Kopyalanan ya da kesilen bir bölümü, istenilen yere taşımak/yapıştırmaktır.

Format: Yazdığımız dökümanın şekillendirilmesine, düzen verilmesine format adı verilir.

Find: Arama olarak tanımlanır. Yazdığımız döküman içerisinde herhangi bir karakter dizisinin tüm metin içerisinde program tarafından bulunmasıdır.

Replace: Find gibi işler. Aranan sözcük ya da sözcük grubu, verilen yeni bir sözcük ya da sözcük grubu ile değiştirilebilir.

Font: Yazı karakterlerinin çeşitli biçimleridir. Yazı tipidir. Dökümanımıza estetik bir görünüm vermek için çeşitli yazı karakterleri vardır. Bunların boyları da ayarlanabilir. İşte bu karakterlere **font** adı verilir.

Get: Daha önce yazdığımız bir dosyayı tekrar belleğe çağırmadır.

Save: Bir dökümanı bellekten disk/diskete kayıt etmektir.

Continue: Devam anlamındadır.

Cansel: Çıkış anlamındadır.

Diğer terimleri **PW** editörünü kullanırken sırası geldikçe öğreneceğiz.

E. PROFESSIONAL WRITE (PW) PROGRAMINA

GİRİŞ

PW kelime işlem programı kullanım kolaylığı ve bir daktilodan daha gelişmiş yazım yapabildiği için piyasada en çok kullanılan kelime işlem programlarından biridir.

Bir ana menü ve alt menülerden oluşur. Döküman yazımı sırasında alt menülere ulaşılır. Bu menülere fonksiyon tuşları ile geçilir. **PW**'de bazı kelime işlem programında olmayan çizim modu da vardır. Çizim modu ile dökümanlarımıza daha iyi biçim vermek için, tabloların kenarlarına çerçeve çizebiliriz.

PW programını çalıştırabilmek için, bir diskete programı yüklemek gerekir. Bundan sonra programı çalıştırabilmek için yapacağımız tek şey **PW** yazıp **Enter** tuşuna basmaktır. Bu işlemden sonra **PW** yüklenecek ve ekrana **Ana Menü** gelecektir.

PW programını bir dizin içerisine yükleyerek, diğer programlarla karışmasını önlemek en iyi yoldur. **PW** programının bulunduğu sürüçüye geçerek, dizin içerisine gireriz. Örneğin **PW** adlı bir dizin açmış isek:

```
CD PW ↵
```

Yazıp **Enter** tuşuna basarız. Böylece **PW** dizini içerisine girmiş oluruz.

PW yazıp **Enter** tuşuna basarak **PW** programını çalıştırabiliriz.

F. ANA MENÜ

MAIN MENU

PW programını çalıştırdıktan sonra karşımıza ilk olarak **Ana Menü** (Main Menu) çıkacaktır.

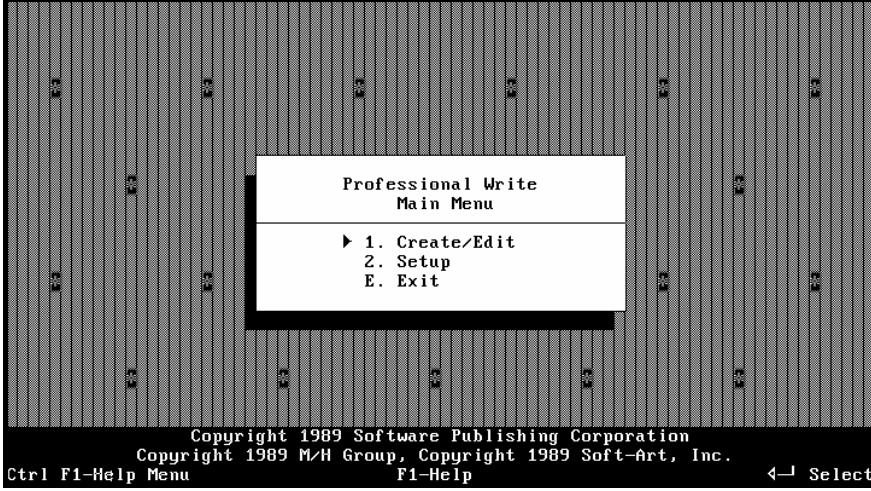
Ana Menü ile **PW**'ye girerek çalışma yapabilir, programı bilgisayarımıza uyarlayabilir, kayıt birimini ve yazıcımızı tanıtabilir, programdan çıkabiliriz.

Ana menü'de görülen seçeneklerin anlamları kısaca şöyledir:

1. Create / Edit: **PW**'de çalışma yapmak için seçim yapılır. Yani çalışma alanına girilir.

2. Setup: **PW** kelime işlem programına bilgisayarımızı ve yan birimlerini tanıtmak için kullanılan bölümdür.

E. Exit: **PW** programından çıkarak **DOS** ortamına dönüş içindir.



Yukarıda gördüğünüz **Ana Menü** ekranında üç seçenek vardır. Bu seçeneklerden istediğimiz birini seçmek için, ya ışıklı göstergiyi ok tuşları ile seçeceğimizin üzerine getirip **Enter** tuşuna basarız ya da seçeceğimiz bölümün sol yanında bulunan harf/rakam'a basarız.

1. YARAT/DÜZELT

Create/Edit

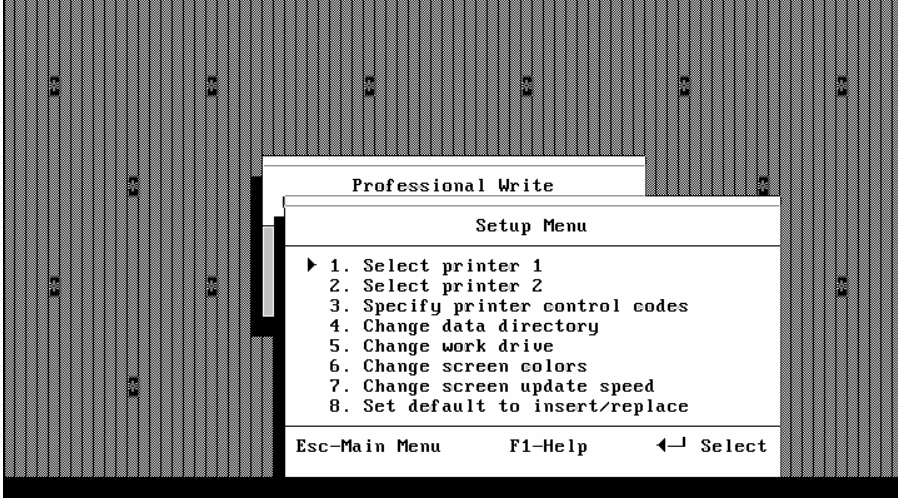
Yarat/Düzeltil bölümüdür. Bu bölümü seçtiğimizde ekran değişecektir. Karşımıza gelen ekran yazım ekranıdır. **WORKING COPY** alanı olarak tanımlanır. Yeni bir dosya yaratmak, daha önce kayıtt ettiğimiz bir dosya üzerinde çalışma ve düzeltme yapma işlemi bu bölümde gerçekleştirilir. Çıkış için **Escape (ESC)** tuşuna basarak **Ana Menü**'ye geçebiliriz.

2. DÜZENLE

Setup

PW programına bilgisayarı ve yan üniteleri tanıma bölümüdür. Yazıcı, kayıt ortamı, renk ayarı vb. tanıtılarak düzenleme yapılır. Bu düzenlemeler yapılmadığı takdirde, tanıtılmayan yan üniteler ile program arasında iletişim eksikliği olacağından istediğimiz sonucu alamayabiliriz. **Setup**'da yaptığımız değişiklikler, kalıcı olarak kaydedileceğinden bir kez yapmamız yeterli olacaktır. Yan ünitelerde değişiklik olması ya da **PW**'nin yeniden kurulumu söz konusu olduğunda yeniden setup ayarları güncelleştirilir.

Seçim sonrası karşımıza şu ekran çıkacaktır:



1. BİRİNCİ YAZICI SEÇİMİ

Select Printer 1

Yazıcı tanımlamamızı bu bölüm aracılığı ile yaparız. **PW** tarafından kabul gören yazıcıların listesi içerisinde yazıcımıza uyan seçeneği seçerek yazıcı tanımlamamızı yapmış oluruz. Birden çok yazıcı kullanıyorsanız, en çok kullandığımız yazıcıyı seçmeniz çıktılarımızı o yazıcıdan almanızı sağlar.

Printer 1 Selection		
Printers		Printer Ports
Epson LQ-2550	HP RuggedWriter	PRN:
Epson LX-80	HP ThinkJet	LPT1:
Epson LX-86/800	IBM Color Jetprinter	LPT2:
Epson MX-80/100 III	IBM Color Printer	LPT3:
Epson RX-80/100	IBM Graphics	AUX:
Fujitsu DL2400	IBM Pers. Pageprintr	COM1:
Fujitsu DX2000 Ser.	▶ IBM Pro. II/KL	COM2:
HP DeskJet	IBM Pro. X24/KL24	
HP LaserJet	IBM Pro. X24E/KL24E	
HP LaserJet Plus	IBM Quickwriter	
HP LJ Series II	IBM Quietwriter	
HP LJ Series IID	IBM Quietwriter III	
HP PaintJet	IBM Wheelprinter	
HP QuietJet	Kyocera F-Series	
Printer: IBM Pro. II/KL		
Printer port: LPT1:		
Select printer and printer port.		
Esc-Cancel	Tab-Select port	F1-Help PgUp,PgDn-More printers ← Continue

Not: Eğer kullandığımız yazıcı PW'nin tanıdığı yazıcılar arasında yoksa **IBM Proprinter** yazıcısını seçmeyi deneyin. Çoğu yazıcılar **IBM Proprinter**'i emüle ederler.

2. İKİNCİ YAZICI SEÇİMİ

Select Printer 2

İkinci bir yazıcı kullanıyorsanız, bu yazıcıyı tanımlamanızı sağlar.

3. YAZICI KONTROL KODLARINI BELİRLEME

Specify Printer Control Codes

Ekranda iki yazıcı için kontrol kodu tanımlamaları çıkacaktır. Döküman için sayfa öncesi ve sonrası istediğiniz kadarını tanımlayabilirsiniz. Hepsini tanımlama zorunluluğu yoktur.

Printer Control Codes		
Printer 1		
Before document :		
After document :		
Before page :		
After page :		
Printer 2		
Before document :		
After document :		
Before page :		
After page :		
Enter printer control codes.		
Esc-Cancel	F1-Help	← Continue

4. VERİ KAYIT ALANINI TANIMLAMA

Change Data Directory

Yazdığımız dökümanların kaydedileceği **sürücü** ve **dizin** (directory) belirtilerek kayıtların belirttiğimiz ortama yapılmasını sağlarız.

Current data directory:		
Type pathname for default data directory.		
Esc-Cancel	F1-Help	← Continue

Örnek:

- A:\PW** Yazarsak, dökümanlarımızı **A** sürücüsündeki **PW** dizini içerisine kaydetmesini istemiş oluruz.
- C:\PW\DATA** Yazarsak **C** sürücüsünde bulunan **PW** dizininin alt dizini olan **DATA** dizinine kayıt edilmesini sağlarız.

5. ÇALIŞMA SÜRÜCÜSÜ TANIMLAMA

Change Work Drive

PW programı ile çalışacağımız sürücü bu bölümde tanıtılır.

```
Current work drive:
Type name of work drive.
Esc-Cancel  F1-Help  ← Continue
```

Örnek:

- A** sürücüsünde çalışma yapılacaktır.
- C** sürücüsünde çalışma yapılacaktır.

6. EKLAN RENKLERİNİ TANIMLAMA

Change Screen Colors

Renkli ekran kullanıyorsak bu bölümde, geri planda kalan renkleri ve yazım renklerini değiştirerek istediğimiz renkte ekran görünümü elde edebiliriz.

```
Change Screen Colors
1. Color scheme #1
2. Color scheme #2
▶ 3. Color scheme #3
4. Monochrome
```

7. EKCRAN GÖSTERİM HIZINI AYARLAMA

Change Screen Update Speed

Ekran gösterim hızını ayarlamamızı sağlar. Ekranınız hızlı ise ayarlama yapmaya gerek yoktur. Zaten böyle bir durumda ayarlama seçeneđi çıkmaz. Yavaş ise **Fast screen update (Y/N)**: şeklinde bir mesaj çıkar. **Y** seçeneđini girerek hızı güncelleyebilirsiniz.

8. INSERT TUŞUNU AÇMA/KAPAMA

Set default to insert/replace

Insert/Replace (I/R): I
Select default typing mode. Esc-Cancel F1-Help ← Continue

I yazarsak **PW** açıldığında **Insert** modunda açılır. **R** yazarsak **Insert** modunda açılmaz, *üzerine yazma moduna* geçilir.

3. ÇIKIŞ

Exit

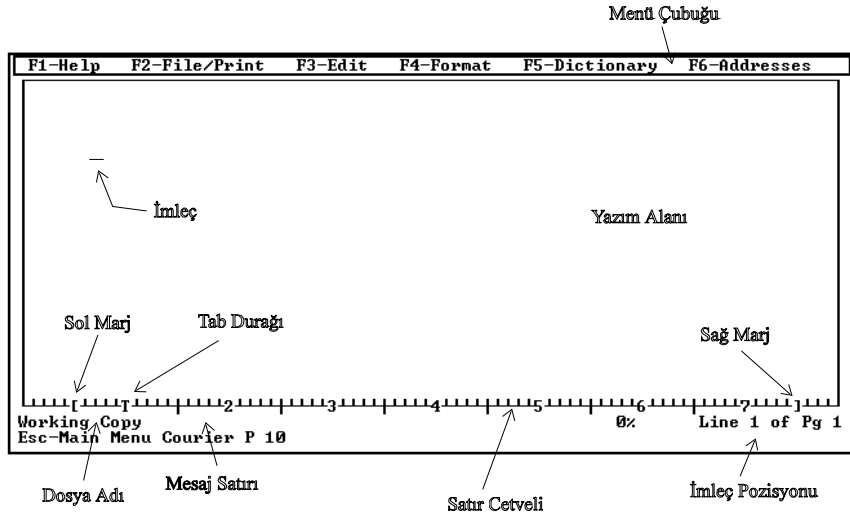
PW programından çıkmak, **DOS** ortamına dönmek içindir. **E** tuşuna basarak ya da ışıklı göstergeyi **EXIT** yazısının üzerine ok tuşları ile getirip **Enter** tuşuna basarak **PW**'den çıkarız.

W A R N I N G
You are about to exit Professional Write. Your changes have not been saved.
Esc-Cancel ← Continue

PW'de yazdığımız son dökümanı kayıt etmeden çıkıyorsak bu mesaj ile uyarılırız. Tekrar dönüp kayıt etmek için **Esc** tuşuna, çıkmak için **Enter** tuşuna basarız.

G. WORKING COPY ALANI VE KLAVYE TUŞ ÖZELLİKLERİ

Çalışma kopyalama alanıdır. **1. Create/Edit** seçeneği ile yazım alanına girdiğimiz bölümdür. Yazım sırasında yapacağımız bütün işlemler bu bölümde yapılır.



Çalışma ekranı karşımıza geldiğinde ekran değişir. Üst kısımda alt menülere geçebilmek için fonksiyon tuşlarının görevleri vardır.

Alt kısım ise durum cetveli olarak adlandırılır. **Durum Cetvelinde** (Satır cetveli) kaçınıcı satır, kaçınıcı kolonda bulunduğumuzu, tab yerlerini, çalışma dosyası adını vb. görebiliriz. Çıkış için **Escape (Esc)** tuşuna basarak **Ana Menü**'ye geçebiliriz.

YARDIM MENÜSÜ

F1 HELP Menu

Fonksiyon tuşlarından **F1** Tuşu, **PW** programı hakkında yardım almak için kullanılır. Yardım menüsü sayesinde kullanım sırasında karşılaştığımız sorunlarda yardım menüsünden, kullanım hakkında yardım alabiliriz.

Cursor Movement Keys	
Press:	To move to the:
Ctrl-Left Arrow	Previous word
Ctrl-Right Arrow	Next word
Home	Beginning of line
End	End of line
Ctrl-PgUp	Previous document page
Ctrl-PgDn	Next document page
Ctrl-Home	Beginning of document
Ctrl-End	End of document
Ctrl-J	Page you want to jump to
Tab	Next tab stop
Esc-Cancel	↓,PgDn-More
	Ctrl F1-Help Menu

F1 tuşuna boş ekranda bastığımızda karşımıza, PW editöründe tuşların kullanımı hakkında yardım çıkar.

PW ile çalışırken, yardım almak istediğimizde, hangi bölümde bulunuyorsak o bölüm ile ilgili yardım alabiliriz.

PW yazımı sırasında klavye kullanımı için bazı tuşlara değişik görevler yüklenmiştir. Bu özellikler PW programını rahat ve kolay kullanmamızı sağlar.

PW Editör'ünde klavye tuşlarının kullanımı şu şekildedir:

İkili Tuş Kullanımları

Tuşlar	Anlam ve görevleri
Ctrl + ←	Bir önceki sözcüğe geçer.
Ctrl + →	Bir sonraki sözcüğe geçer.
Ctrl + Home	İmleci dökümanın başına götürür.
Ctrl + End	İmleci dökümanın sonuna götürür.
Ctrl + Page Up	Bir önceki sayfaya gider.
Ctrl + Page Down	Bir sonraki sayfaya gider.
Shift + Tab	Her basışta bir önceki Tab noktasına gider.

Edit Tuşlarının Tek Başına Kullanımları

Tuşlar	Anlam ve görevleri
Esc (Escape)	Cansel anlamındadır. Geri dönüş ve çıkış için kullanılır.
Delete	İmlecin üzerinde bulunduğu karakteri silerken, imleç yerinde kalır ve sağındaki karakterleri kendine doğru çeker.
Insert	Araya ekleme moduna geçilir. Aşağıda Inserting yazısı çıkar. İkinci basışta üzerine yazma moduna geçer. Yani Insert modundan çıkılır.
Home	İmleci satırın başına getirir.
End	İmleci satırın sonuna götürür.
Tab	Her basışta imleci bir sonraki Tab noktasına götürür.
Page Up	Her basışta bir önceki ekranı getirir.
Page Down	Her basışta bir sonraki ekranı getirir.
←	Her basışta imleci bir karakter sola götürür.
→	Her basışta imleci bir karakter sağa götürür.
↑	Her basışta imleci bir satır yukarı götürür.
↓	Her basışta imleci bir satır aşağı götürür.

Bu tuşların yanı sıra **PW** Menülerini rahat kullanmak için de, bazı tuşlara değişik görevler yüklenmiştir. Bu tuşları menüleri kullanırken sırasıyla göreceğiz.

Dosya İçine Eklenecek Komutlar

NEW PAGE	ya da	*n*	Yazım sırasında yeni bir sayfaya geçer.
JOIN	ya da	*j*	Dosyayı yazılan dökümanın parçası olarak yaz.
PRINTER	ya da	*p*	Yazıcıya kontrol kodlarını gönder.
GRAPH	ya da	*g*	Grafiği yazılan dökümanın parçası olarak yaz.
JUSTIFY YES	ya da	*jy y*	Sola/sağa dayalı yazdır.
JUSTIFY NO	ya da	*jy n*	Sola/sağa dayalı ayarı kapa.

Durum Cetvelinde Görülen İşaretlerin Anlamları:

- İmlecin sütun durumunu verir.
- [] Sol ve Sağ marjları gösterir. Yazım sınırları.
- T ya da D Tab yerlerini gösterir.
- > Geçici olarak ayarlanan sayfa solu ayarını gösterir.

Durum Cetvelinin Altında Çıkan Mesajların Anlamları:

Working Copy: Yeni bir dosya üzerinde çalışıyorsak bu mesaj çıkar. Daha önce kayıt ettiğimiz bir dosya ise, üzerinde çalıştığımız dosyanın adı yazılacaktır.

Inserting: Ins (Insert) tuşuna basılı olup olmadığını gösterir. Bu mesaj varsa **Insert modundadır**. Yani araya girme modundadır. Bu durumda imleç yazının üzerinde ise, basılan tuşlar araya eklenerek yazıların sağa kayması sağlanacaktır. **Inserting** yazmıyorsa, tuşlara basılınca üzerine yazar.

Drawing: Çizim modunda olduğumuzu belirtir.

Bold: İmleç koyu yazı yazdığımız sözcüklerin üzerine geldiğinde **Bold** yazısı çıkar.

Under: İmleç alt çizgi için işaretlediğimiz sözcüklerin üzerinde iken çıkar.

Italic: İmleç italik yazı karakteri olarak işaretlediğimiz yere geldiğinde çıkar.

x%: PW'nin izin verdiği kullanılabilen belleğin yüzde kaçının kullanıldığını gösterir.

Line xx of Pg xx: **Line xx** kaçınıcı satırda olduğumuzu, **Pg xx** kaçınıcı sayfada olduğumuzu belirtir.

Esc-Main Menu: Esc tuşuna basıldığında **Ana Menü**'ye döneceğini belirtir.

Bunların dışında, yazıcımızın font desteği varsa, kullandığımız fontun tipi de durum cetvelinin altında yazılacaktır.

Alt menülere geçmeden önce alt menülerde görülen komutların nasıl kullanılacağını görelim:

Alt menüleri seçmek için üzerinde yazılı olan fonksiyon tuşlarına basmak gerekir.

Menü seçildikten sonra karşımıza komutlar penceresi gelecektir. Bunları şu şekilde kullanabiliriz:

Menü seçimi yapıldıktan sonra kullanacaksak, kullanacağımız komutu belirledikten sonra ya sol yanında belirtilen rakama/harfe basarak seçeriz ya da ışıklı gösterge ile birlikte beliren üçgen biçimindeki imleci ok tuşları ile seçeceğimiz komutun üzerine getirerek **Enter** tuşuna basarız.

Menüye geçmeden de bazı komutları kullanabiliriz. Bu şekilde kullanabileceğimiz komutların sağlarında belirtilen tuşlara bastığımızda, menü seçimi yapmadan kısa yoldan istediğimiz komutu vermiş oluruz. Bu şekilde kullanılan tuşlara **kolay tuşu** adını veriyoruz.

Bir örneği birlikte yapalım:

Bir sözcüğü kalın yazdırmayı deneyelim. İmleci koyulaştırmak istediğiniz herhangi bir sözcüğün üzerine getirin. Sonra **F3** tuşuna basarak **Edit Menü**sünü seçelim. Bu menüde gördüğümüz **7.** seçenekte **Boldface Word**, sözcük koyulaştırmak içindir. Şimdi klavyedeki **7** tuşuna basalım. Gördüğümüz gibi sözcük koyulaştı.

Aynı şeyi bir kez daha başka yoldan yapalım. Yine imleci herhangi bir sözcüğün üzerine getirelim ve **F3** tuşuna basalım. **7**'yi seçmek için ışıklı göstergeyi aşağı ok tuşu ile **7.** seçeneğin üzerine getirelim ve **Enter** tuşuna basalım. Gördüğümüz gibi sözcük yine koyulaştı.

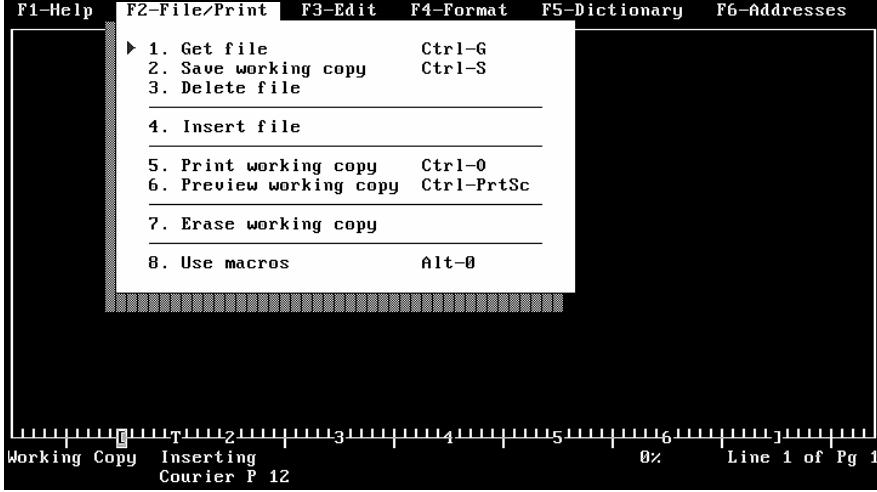
Sözcük koyulaştırma işini bu kez de daha kısa yoldan yapalım. **F3** menüsünü seçtiğimizde **7. Boldface Word** seçeneğinin karşısında **Ctrl-B** tuş kullanımı yazıyordu. İmleci yine bir sözcüğün üzerine getirelim ve menüye geçmeden **Ctrl+B** tuşlarına birlikte basalım. Yine belirlediğimiz sözcük koyulaştı.

Bu örneğimize dayanarak bundan sonraki menülerin anlamlarını öğrendikten sonra, artık kolayca kullanabiliriz.

H. DOSYA ve BASKI İŞLEMLERİ

F2 File/Print

F2 Alt Menüsü, F2 tuşuna basarak seçilir. F2 Alt Menüsü, dosya ve baskı işlemleri için kullanılır.



Yukarıda gördüğümüz F2 Alt Menüsü ekranında görülen komutların anlamları şöyledir:

1. DOSYA ÇAĞIRMA

Get file (Ctrl-G)

Filename :

Esc-Cancel ← Get

Filename: Çağırılacak dosyanın sürücü ve yol belirtilerek çağırılmasını sağlar. Yanına sürücü, dizin ve dosya adı yazılır.

Diskte/diskette olan bir dosyayı çağırmak için kullanılır. Dosyayı çağırmak istediğimizde yeni bir ekran gelir. Bu ekranda **Filename: Sürücü:** ve **setup'da** yol belirtmişsek yol adı çıkar. Buraya çağırmak istediğimiz dosya adını yazarak çağırabiliriz.

Örnek:

A: \PW\SINAV Şeklinde yazarak, A sürücüsündeki PW dizininde bulunan SINAV adlı dosyayı çağırabiliriz.

C: \PW\DATA\ALACAK.DAT Şeklinde yazarak, C sürücüsündeki PW dizininin alt dizini olan DATA dizininde bulunan ALACAK.DAT adlı dosyayı çağırabiliriz.

Bir başka yol da, dosya adı yazmadan **Enter** tuşu ile boş geçerek yeni bir ekranın gelmesini sağlayıp burada görülen dosyalardan birini ışıklı gösterge aracılığı ile seçerek çağırabiliriz. Eğer dosyalar çok ve bir ekrana sığmıyorsa **Page Up/Page Down** tuşuna basarak önceki/sonraki sayfaları görerek seçim yapabiliriz.

İmleç Pozisyonu

Bulunulan Ortam

Üst Dizin

Directory Listing of a:\PW				
Filename	Ext	Date	Size	Description
▶ ..		6/20/94	0	Parent Directory
DATA		6/20/94	0	Directory
KITAP		6/20/94	0	Directory
OZEL		6/20/94	0	Directory
PFS	.DIR	1/01/80	236	PFS directory catalog file.
PW	.COM	2/04/88	4051	
PW	.HLP	2/04/88	22409	
PW	.LML	1/02/80	52080	
PW	.MAC	3/11/93	1834	
PW	.NET	4/20/93	60	
PW	.PIF	2/04/88	369	
PW	.PRG	2/04/88	318505	
PW	.PRI	2/04/88	11396	
PW	.SET	4/09/94	1470	

Alt Dizin

Directory or filename: ..

Esc-Cancel F1-Help F8-Sort Get file. F9-Search documents ← Get

Dizin ve Dosya Adı

F8: Dosya adlarını sort eder. Harf sırasına göre dizer.

F9: Döküman aramak için kullanılır.

Eğer çağırdığımız dosya daha önce PW ile yazılmamışsa bizden dosya çağırma sırasında dosyanın hangi editörle ya da hangi formatta yazıldığı sorulur. Bunu belirleyip ona göre seçim yapmamız gerekir. Değilse bizim yazı dilinde kullanmadığımız karakterle karşılaşırız. Çağıracağımız dosya bir ASCII dosya ise bunu belirterek seçmeliyiz vb. Bir dosyayı çağırırken eğer bellekte bir çalışmamız varsa bunu kayıt etmemişsek PW tarafından uyarılacaktır.

W A R N I N G	
You are about to overwrite the working copy. Your changes have not been saved.	
Esc-Cancel	← Save

Bu uyarı mesajı, bellekte bir dosya var iken yeni bir dosya çağırıldığında, daha önceki dosyanın kaydedilmediği durumlarda çıkar. Eğer kayıt etmek için geri döneceksek **Esc** tuşuna basarız. Kaydetmeyeceksek, **Enter** tuşuna basarak istediğimiz dosyanın belleğe yüklenmesini sağlarız.

2. DOSYA KAYDETME

Save working copy (Ctrl-S)

PW ile yazdığımız dökümanlar geçici olarak bellektedir. Bu dökümanları daha sonra kullanabilmemiz için çevre bellek birimlerinden birine kayıt etmemiz gerekir.

Yazdığımız dökümanı disk/diskete kayıt etmek istediğimizde kullanılır. Belirli aralıklarla kayıt etmemiz, dökümanımızı elektrik kesilmeleri vb. durumlardan korumak açısından yararlıdır.

Kayıt seçiminden sonra ekrana gelen tablo bize kayıt araçlarını ve biçimini belirlememizi sağlar.

Filename :	
Description :	
File type (Normal/ASCII/Encrypt/Other) :	N
Esc-Cancel	← Save

Bu seçeneklere geçmek için **Tab** tuşu kullanılır.

Tabloda geçen terimlerin anlamları:

Filename: Kayıt edilecek dosyanın sürücü ve yol belirtilerek kayıt edilmesini ve dosyaya bir ad vermemizi sağlar. Yeni yazdığımız bir dosya ise bir ad veririz. Eğer dosyamız eski ise zaten burada ad olduğundan gerek kalmayacaktır. İstersek burada yeni bir sürücü, yol ve dosya adı vererek kayıt yapabiliriz.

Description: Dökümanla ilgili tanımlama yapılır. En çok 40 karakter olabilir.

File type (Normal/ASCII/Encrypt/Other): Dosyanın kayıt biçimini belirler. İmleci buraya taşıyarak, kayıt biçimi için seçiminizi kayıt seçeneklerinin ilk harflerini kullanarak yaparız.

Kayıt seçenekleri

Normal (N)	Bu kayıt biçimini seçmek için N yazılır. PW editörünün kendi kayıt biçimidir. Kayıt sırasında PW editörünün kontrol karakterleri ile kayıt edileceğinden, dökümanımıza verdiğimiz biçim bozulmadan korunacaktır.
ASCII (A)	ASCII kayıt etmeyi sağlar. A harfi seçilmelidir. Bu kayıt biçimini seçtiğimizde PW editöründe hazırladığımız biçim kaybolur.
Encrypt (E)	Kayıt edilecek dosyanın şifreli kayıt edilmesi içindir. Tekrar ulaşmak istediğimizde şifreyi doğru yazmamız gerekir.
Other (O)	Dosyayı başka bir yazılıma aktaracaksa bu kayıt biçimini kullanırız. Bu seçimden sonra yeni bir tablo gelir ve bize hangi yazılıma uygun kayıt edileceğini belirlememizi ister.

Esc - Cansel Kayıt işleminden geriye dönüş için Esc tuşuna basarız.

Daha önce kaydettiğimiz bir dosya var ve aynı adla yeni bir kayıt yapacaksa PW bizi aşağıdaki mesajla uyarır:

W A R N I N G
You are about to overwrite an existing file.
Esc-Cancel ← Continue

3. DOSYA SİLME

Delete file

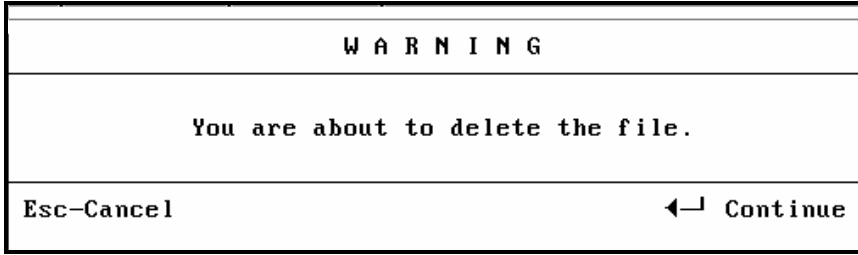
Disk/diskete kayıt ettiğimiz dosyaları silerken kullanılır.

Bu seçimden sonra karşımıza gelen tabloda, **Filename:**'in karşısına silmek istediğimiz dosyanın sürücüsü, yolu ve adı belirtilir. Ya da **Enter** tuşuna basılarak dosya listelerini görerek buradan da seçim yapabiliriz.

Örnek:

- A:\PW\BORC.DAT** Yazarak, A sürücüsündeki **PW** dizini içerisinde bulunan **BORC.DAT** adlı dosyayı siler.
- C:\PW\DATA\MEKTUP** Yazarsak, C sürücüsündeki **PW** dizininin alt dizini olan **DATA** dizini içerisinde bulunan **MEKTUP** adlı dosyayı siler.

Dosya silme sırasında, yazılım tarafından silmek isteyip istemeyeceğimiz uyarılacaktır. Burada aşağıdaki mesaj çıkar:



4. DOSYA EKLEME

Insert file

Yazdığımız dökümana, daha önce yazdığımız bir dökümanı eklemek istediğimizde kullanılır.

Eklemek istediğimiz dökümanı, üzerinde çalıştığımız dökümanın neresine ekleyeceksek imleci oraya getirip **Insert file**'i seçeriz. **Enter** tuşuna bastıktan sonra karşımıza gelen tabloda **Filename:**'in karşısına eklemek istediğimiz dosyanın sürücüsünü, yolunu ve adını belirterek **Enter** tuşuna basacağımız gibi, **Enter**'le boş geçerek dosya listesini görerek de seçim yapabiliriz.

5. YAZICIYA YAZDIRMA

Print working copy (Ctrl-O)

Dökümanımızın tamamını ya da belirlenen bir bölümünü yazıcıya yazdırmak istediğimizde kullanılır.

Print working copy seçeneğini seçtiğimizde karşımıza gelen menü yazıcıya aktarma biçimini belirlememizi sağlar.

```
Print Options
-----
Number of copies:      Starting page:      Ending page:
Pause between pages (Y/N) :
Document or envelope (D/E):
Data file to merge:
Indent                :
Print style:
1. Normal              3. Compressed
2. Justified           4. Letter-quality
                    5. Font format
Print to:
1. HP LJ Series II    3. Disk file
2. IBM Graphics
Download soft fonts (Y/N):
-----
Esc-Cancel  FB-List fonts  Ctrl PrtSc-Preview  ←-Continue
```

Bu menü **Print Options** olarak adlandırılır. Yapmak istediğimiz değişiklikler için **Tab** tuşunu kullanırız. **Shift+Tab** tuşları ile geriye doğru gider. Bu menüdeki terimlerin anlamları şöyledir:

Number of copies	Yazılacak dökümanın, yazıcıdan kaç kopya alınacağı sayısal olarak girip belirlenir.
Starting page	Yazılacak dökümanın kaçınıcı sayfadan başlayacağı belirtilir.
Ending page	Yazılacak dökümanın kaçınıcı sayfada son bulacağı belirtilir.
Pause between page (Y/N)	Dökümanın yazımı sırasında sayfa bitiminde bekleme yapılıp yapılmayacağı belirtilir. Y girilirse bekleme yapılır, N girilirse bekleme yapılmaz.

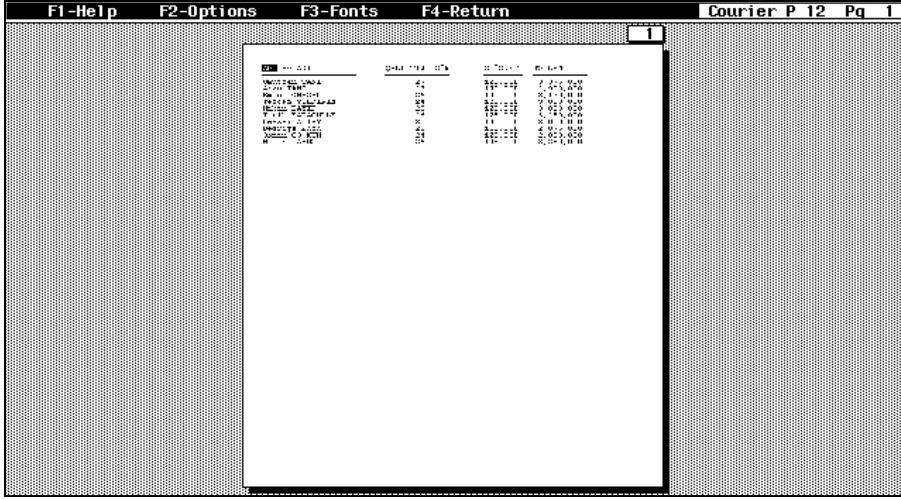
Document or envelope (D/E)	Yazılacak döküman kağıda yazılacaksa D , zarf üzeri ise E seçilir.
Data file to merge	Data dosyası ile birleştirilir.
Indent	Yazıcılar genellikle hemen soldan yazmaya başlar. Kaç karakter içeriden yazılacağı burada belirtilir.
Print style	Yazılacak dökümanın, yazım modunu belirler. Aşağıdaki seçimlerden aynı anda birden fazlasını da seçebiliriz. Bunun için aralarına (,) virgül koyamamız gerekir. 2,3 gibi... Burada yapacağımız seçimlerin anlamları şöyledir: 1. Normal Normal yazım modu. 2. Justified Yazılan dökümanın sola ve sağa bloklanmasını sağlar. 3. Compressed Yazım sıkıştırılmış modda yapılır. 4. Letter-quality Mektup kalitesinde yazım yapmayı sağlar. Yazımın üzerinden iki defa geçerek, daha net çıktı alınır. 5. Font format Yazıcının font desteği varsa, belirtilen fontta yazar.
Print to	Ana Menü'de yaptığımız printer seçimlerinden hangisine gönderileceği belirtilir.
F8	Seçtiğimiz yazıcıya uyan font listesini verir. Buradan font seçimi yapılarak, yazıcıdan istenilen fontta çıktı alınabilir.
Ctrl PrtSc-Preview	Baskı önizleme. Yazılacak dökümanın yazıcıdan nasıl çıkacağını ekranda gösterir.

Bu seçimler yapıldıktan sonra **Enter** tuşuna basarak yazıcıya gönderilir.

6. BASKI ÖNİZLEME

Print preview (Ctrl-Prt Scr)

Bu bölümde belgelerin kağıda nasıl çıkacağı ekranda gösterilir. **F2** alt menüsünden **6. Print preview** seçeneği seçilir ya da **Ctrl+PrtScr** tuşlarına basılır. Aşağıdaki ekran gelir:



Görüldüğü gibi, yazıcıdan kağıda nasıl aktarılacağı ekranda çıkar.

Ekranın üst kısmında fonksiyon tuşlarına çeşitli görevler yüklenmiştir. Bu görevleri sırasıyla görelim:

a. F1-Help

Bölümle ilgili yardım verir. Burada da bulunduğunuz bölümde iken bastığımızda size o konuyla ilgili yardım verecektir.

b. F2-Options

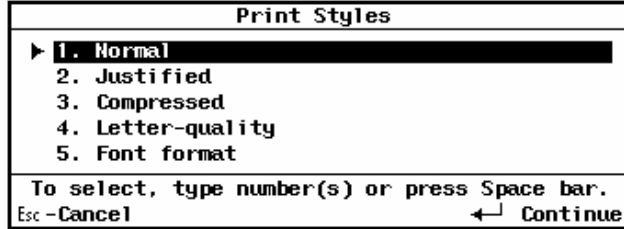
Baskı önizleme opsiyonları bu bölümedir. Buradan ekranı büyütülebilir, yazı stili seçilebilir ve kağıt boyunu ayarlayabilirsiniz. F2 seçildiğinde aşağıdaki menü seçimi gelir:



1. Change view Alt V: Görüntünün ekrana sığdırılması sağlanır. Aşağıdaki görüntü gelir:

ADI SOYADI	ÇALIŞTIĞI GÜN	G. ÜCRET	TOPLAM
Umutcan TANIS	27	125,000	3,375,000
Arzu TANIS	23	130,000	2,990,000
Emine ONACAK	26	120,000	3,120,000
Tezcan YILDIRIM	24	125,000	3,000,000
Hakan KAYIR	25	120,000	3,000,000
Tarik KARAGÜLLE	26	125,000	3,250,000
Kevser ALTAY	20	100,000	2,000,000
Bedriye KAYA	25	115,000	2,875,000
Osman COSKUN	24	120,000	2,880,000
Hasan YAVUZ	26	125,000	3,250,000

2. Set print styles Alt S: Yazı stili seçimini sağlar. Aşağıdaki pencere gelir:



Seçim için ya ok tuşlarından yararlanırız, ya da sol tarafındaki rakamlara basarız.

3. Change paper size Alt P: Kağıt seçimi yapılır. İstenilen boyutta kağıt seçmek için, ok tuşları ile üzerine gelinir ve Enter tuşuna ya da sol taraftaki rakam tuşuna basılır. aşağıdaki ekran gelir:

Paper Size	
1. Letter (8½" x 11")	
2. Legal (8½" x 14")	
3. A4 (210 mm x 297 mm)	
4. B5 (176 mm x 250 mm)	
Esc - Cancel	← Select

c. F3-Fonts

Font seçimi menüsüdür. Aşağıdaki ekranda görülen seçenekleri sunar:

F1 - Help	F2 - Options	F3 - Fonts	F4 - Return	Courier P 12 Pg 1
		1. Change a font Alt F		
		2. List fonts in document Alt D		
		3. List available fonts Alt A		

1. Change a font Alt F: Font değiştirmek ve boylarını ayarlamak için bu seçenek kullanılır, aşağıdaki ekranda görüldüğü gibi font listesi gelir:

Select Font or Size			
Font	Orientation	Point	Pitch
LJ11D-Courier	Portrait	12	10
LJ11D-Courier	Landscape	12	10
LJ11D-Courier	Portrait	10	12
LJ11D-Line Printer	Portrait	8.5	16.66
LJ11D-Line Printer	Landscape	8.5	16.66
B-HELW bold	Portrait	14.4	Prop.
B-TMS RMN	Portrait	10	Prop.
B-TMS RMN light	Portrait	8	Prop.
B-Line Printer light	Landscape	8.5	16.66
Esc - Cancel	PgDn - More		← Select

İstenilen font için ok tuşları ile üzerine gelinir Enter tuşuna basılarak seçim yapılır. Seçim sonrası değiştirmenin yapıp yapılmayacağı aşağıdaki ekranla uyarılır:

Change the regular font throughout document? (Y/N) <u>N</u>
Esc - Cancel ← Continue

2. List fonts in document Alt D: Döküman içinde kullanılan fontların listesini verir. Aşağıdaki ekran gelir:

Fonts in Document			
Font	Orientation	Point	Pitch
Courier	Portrait	12	10
Esc - Cancel			

3. List available fonts Alt A: Yazıcının desteklediği fontların listesini verir. Aşağıdaki ekran gelir:

Printer 1 Fonts			
Font	Orientation	Point	Pitch
LJ11D-Courier	Portrait	12	10
LJ11D-Courier	Landscape	12	10
LJ11D-Courier	Portrait	10	12
LJ11D-Line Printer	Portrait	8.5	16.66
LJ11D-Line Printer	Landscape	8.5	16.66
B-HELW bold	Portrait	14.4	Prop.
B-TMS RMN	Portrait	10	Prop.
B-TMS RMN light	Portrait	8	Prop.
B-Line Printer light	Landscape	8.5	16.66
Esc - Cancel		PgDn - More	

d. F4-Return

Çıkış ve geri dönüş için seçim penceresi aşağıdaki biçimde gelir:

F1-Help	F2-Options	F3-Fonts	F4-Return	Courier P 12 Pg 1
1. Return to current position Esc				
2. Return to starting position Alt X				

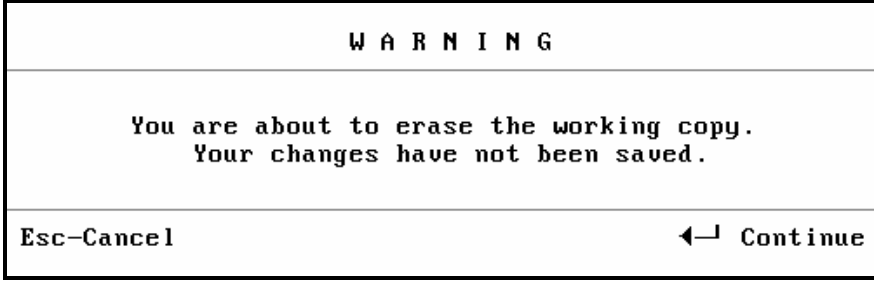
1. Return to current position Esc: Geçici olarak geri dönmek için.

2. Return to starting position Alt X: Yeniden başlamak için geri döner.

7. BELLEĞİ SİLME

Erase working copy

Yeni bir döküman yazmak istediğimizde belleği silmek için kullanılır. Ekranda yazı varsa ve kayıt etmemişsek, dökümanın saklanmadığı için **WARNING** mesajı ile uyarılır.

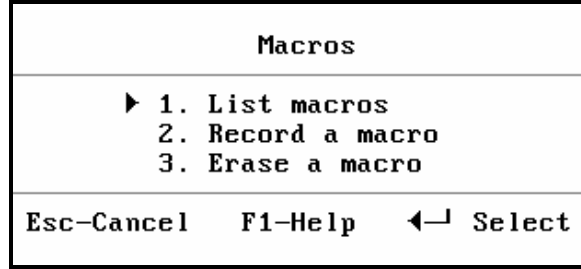


Tekrar **Enter** tuşuna basarsak temizlenir. Vazgeçmek için **Esc** tuşuna basmalıyız.

8. MAKRO KULLANIMI

Use macros (Alt-0)

Yazım sırasında sık sık yinelenen işlemleri, yeniden yapmak yerine makro kullanımı ile tanımlayarak daha kısa sürede yapmamızı sağlar. **Alt-0** tuşuna basarak da geçilir. Seçim sonrası aşağıdaki pencere çıkar:



Makro kullanımı seçildiğinde ekranda yer alan seçeneklerin anlamları şöyledir:

- 1. List macros** Daha önce yapılan makroların listesini verir.

Liste seçimini, daha önce yaptığımız makronun adını ve hangi tuşlara kayıt ettiğimizi anımsamak için kullanabileceğimiz gibi, ışıklı göstereyi ok tuşları ile istediğimiz makronun üzerine getirip **Enter** tuşuna basarak da, ilgili makroyu yazdırabiliriz.

Macro Definitions	
Key	Description
▶ Alt-A	KLISE
Alt-B	BASLIK
Alt-C	
Alt-D	
Alt-E	
Alt-F	
Alt-G	
Alt-H	
Alt-I	

Press PgDn for more macros.
Esc-Cancel ← Select

2. Record a macro Makro yapımına geçilir.

Yeni bir makro yaparken seçeriz. Makroya başladığımız bir pencere aracılığı ile belirtilir.

You are about to begin recording a macro.	
To see your options while you're recording the macro, press Alt-0.	
Esc-Cancel	← Begin recording

Enter tuşuna basarak makroyu yapmaya başlayabiliriz. Makro yapımı sırasında alt kısmında şeritli bir yazı ile **Recording** mesajı, makro yaptığımızı belirtir. Makro bitiminde **Alt-0** tuşlarına basarak yeni bir pencerenin gelmesi sağlanır. Gelen pencerede şu seçenekler vardır:

Recording Options	
▶ 1. Pause the macro for your input	
2. End and save recorded keystrokes	
3. Quit and erase recorded keystrokes	
Esc-Resume recording	← Select

- 1. Pause the macro for your input** Macroyu başlatma, ara verme ve durdurma.
- 2. End and save recorded keystrokes** Makro bitiminde bu seçenikle makroyu kayıt etmeyi sağlarız. Bu seçimden sonra karşımıza Alt tuşları ile hangi tuşu kullanarak makroyu kullanacağımız sorulacaktır. Bunlardan birini seçer ve buna bir isim vererek makro kaydını yapmış oluruz.
- 3. Quit en erase recorded keystrokes** Makro yapmaktan vazgeçtiğimizde bu seçeneği kullanırız. O an bellekte yaptığımız makro silinir.

Bellekte makro yaparken, makro yapımından **3.** seçeneği seçip silerek vazgeçmek istersek aşağıdaki mesaj ile uyarılırız:

W A R N I N G	
You are about to cancel recording of a macro.	
Esc-See recording options	← Cancel recording

Yapmakta olduğumuz makroyu silmek istersek **Enter** tuşuna, vazgeçmek için **Ese** tuşuna basarız.

- 3. Erase a macro** Daha önce yapılan makroların silinmesi.

Bir makroyu birlikte aşama aşama yapalım. Adımızı soyadımızı yazdıralım:

1. **F2** menüsünden **8.** seçeneği ya da direk **Alt+0** tuşlarına basalım.
2. Gelen ekrandan ikinci seçenek olan **Record a macro** seçeneğini seçelim.
3. Bir ekran gelerek makroya başlandığını belirtecektir. **Enter** tuşuna basalım.
4. Yazım alanına geçerez. Adımızı soyadımızı yazalım. Makroda yazmak istediklerimiz bitti.
5. **Alt+0** tuşlarına basalım. Makro listesinin olduğu pencere gelecektir. Biz örnek olarak makromuzu **Alt-D** tuşlarına atayalım. Bunun için **Alt-D** tuşunun olduğu yere gelelim.
6. Makromuza burada bir ad verelim. **ADI** diyelim. **Enter** tuşuna basalım.
7. Yeni bir pencere gelecektir. Buradan **2.** seçenek olan **End and save recorded keystrokes** seçeneğini seçelim ve **Enter** tuşuna basalım. Artık makromuz kaydedilmiştir.
8. Şimdi yazım alanında iken istediğiniz yerde **Alt+D** tuşlarına basın adınız soyadınız çıkacaktır.

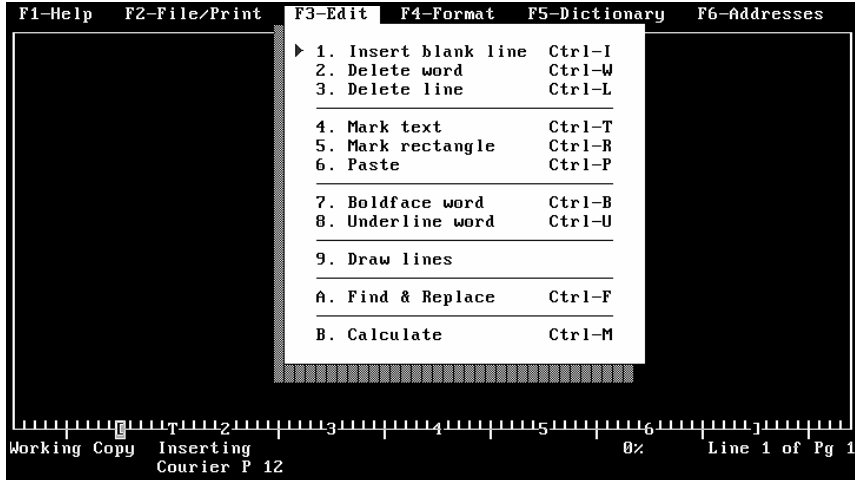
I. DÜZELTME

F3 Edit

Düzeltilme menüsüdür. Çalışma alanında satır, sözcük ve belirlenen bir blok üzerinde silme, kopyalama ve değiştirme gibi düzenleme işlemleri bu bölüm aracılığı ile yapılır.

Bu menü ile çalışmalarımızı istediğimiz düzende yapabiliriz. **PW** menüleri, dökümanlarımızı kullanırken bize çeşitli kolaylıklar sağlar. Bunları sırasıyla göreceğiz.

F3 tuşuna basıldığında ekrana aşağıdaki menü gelir:



1. SATIR EKLEME

Insert blank line (Ctrl-I)

Araya satır eklemek için kullanılır. Bulunulan satırın üzerinde iken **F3** menüsünden ya da **Ctrl-I** tuşlarına birlikte basıldığında yeni bir satır açarak, imlecin hizasındaki satırı aşağıya kaydırır.

2. SÖZCÜK SİLME

Delete word (Ctrl-W)

Sözcük silmek için kullanılır. Silinecek sözcüğün üzerine imleç getirilerek **F3** menüsünden seçilir ya da **Ctrl-W** tuşlarına basılır. Böylece üzerindeki sözcüğü silerek sağındaki yazıları kendine doğru çekecektir.

3. SATIR SİLME

Delete line (Ctrl-L)

Satır silmek için kullanılır. İmleç silinecek satır üzerine getirilir ve **F3** menüsünden seçilerek ya da **Ctrl-L** tuşlarına basılarak satır silinir. İmlecin üzerindeki satırı silerken aşağıdaki satırları kendine doğru çeker.

4. BLOK İŞARETLEME

Mark text (Ctrl-T)

Bir satır ya da bir sözcüğü değil de daha büyük bir bölüm ile ilgili yapılacak işlemler için seçilen bölüme blok dendiğini daha önce söylemiştik. İşte böyle bir bloğun işaretlenmesi gerekmektedir. İşaretlemek istenen bloğun başına imleci getiriniz ve **F3** menüsünden **4.** seçeneği seçerek ya da menüye girmeden **Ctrl-T** tuşlarına basarak blok başını işaretleyiniz. Bu sırada durum cetvelinin altında **F10** tuşu ile ilgili bir mesaj çıkacaktır. Ok tuşları ile bloklamak istediğiniz yere kadar imleci sağa/aşağı götürünüz. Bu işlemi yaparken, işaretlediğiniz yerlerin parlak olduğunu göreceksiniz. İşaretlenen bloğun sonuna gelince **F10** tuşuna basınız. Karşınıza yeni bir pencere açılacaktır. Burada blok için yapmak istediğiniz işlemlerden birini seçiniz. Sonuçta dökümanın tamamının değil yalnızca işaretlediğiniz bloğun yaptığınız işlemin sonucuna göre değiştiğini göreceksiniz.

```

F1-Help  File/Print  Edit  Format  Dictionary  Addresses
ÇOCUKLAR DA ÇINAR OLUR
ey yangın yüzü çocuklar
aktınız bir demet ışıkla
yüzleriniz sonsuz maviliğe karıştı

ordasınız / yüzünüzde bir kan damlası
karlı çınarlar en taze çocukları taşıyor
her bahar aşkı öğrendiğimiz sevinç
yanıyor şimdi sevgilimin teninde

içim bahar gıyınıyor karlar altında
kaçak yağmura verirken gençliğimi
kuramadığım dizeler önüne çıkan
kırık bir tonurcuk danla damla üreyen

yaralı ezgiler savruluyor sessizliğe
ihane aynaları vurgun yemiş
taniyiz golanın kuş kanatlarına
yüzleriniz çiziliyor alınımıza

Güven Tanış

SIIR      Marking text      0%      Line 14 of Pg 1
Esc-Cancel Move the cursor to the end of the block and then press F10.

```

Yukarıdaki örnekte gördüğümüz gibi, bloklanmış alan beyaz bir şerit içerisinde görünmektedir. Bunu nasıl yaptık açıklayalım:

içim sözcüğünün ilk harfi olan **i** harfinin üzerine imleci getirip, **Ctrl-T** tuşuna bastık. Sonra imleci sağa ve aşağı istediğimiz kadar kaydırarak, seçtiğimiz dörtlüğü bloklamış olduk.

Dikkat ederseniz ekranın altında bloklamadan sonra **F10** tuşuna basılması gerektiği mesajı verilmektedir. F10 tuşuna basınca aşağıdaki ekran gelir:

Text Block Operations	
▶ 1. Cut	Del
2. Copy	Ctrl-C

3. Change style	
4. Change font or size	

5. Change left/right margins	Ctrl-[
6. Change tabs	Ctrl-K
7. Change indentation	

8. Single space	
9. Double space	

A. Save	
B. Print	
C. Calculate	Ctrl-M

Esc-Cancel	← Select

F10 Tuşuna bastıktan sonra karşımıza çıkacak işlemlerin anlamları şöyledir:

1. **Cut (Del)** Bloklanan alanı silerek belleğe alır.
2. **Copy (Ctrl-C)** Bloklanan alanı silmeden belleğe alır.
3. **Change style** Bloklanan alanın yazım şeklini değiştirir.

Alt seçenekler Şunlardır:

Style	
1. Boldface	Ctrl-B
2. Underline	Ctrl-U
3. Italic	
4. Superscript	
5. Subscript	
▶ 6. Normal	
To select, type number(s) or press Space bar. Esc-Cancel ← Continue	

1. **Boldface (Ctrl-B)** Bloklanan bölümün yazıcıdan koyu çıkması sağlanır.
2. **Underline (Ctrl-U)** Bloklanan bölümün yazıcıdan altçizgili çıkması sağlanır.
3. **Italic** İtalik yazı (Eğik yazı). Bloklanan bölümün yazıcıdan eğik yazı şeklinde çıkması sağlanır.
4. **Superscript** Bloklanan bölümün yazıcıdan yarım aralık üstte çıkmasını sağlar. Üs işaretleri için kullanılır. X^2 gibi...
5. **Subscript** Bloklanan bölümün yazıcıdan yarım aralık altta çıkmasını sağlar. Alt yazımları için kullanılır. H_2SO_4 gibi...
6. **Normal** İşaretlenen bölümün normal yazımda olmasını sağlar.

- | | |
|---|--|
| 4. Change font or size | Bloklanan bölümün yazı fontlarını değiştirir. |
| 5. Change left/right margins (Ctrl-[]) | Bloklanan alanın sol ve sağ marj ayarlarını değiştirir. |
| 6. Change tabs (Ctrl-K) | Bloklanan alanın Tab ayarlarını değiştirir. |
| 7. Change indentation | Bloklanan alanın satır başını kaydırır. Satır başını kaydırmak için seçim yapıldıktan sonra imleç ok tuşları ile satır başının alınması istenen yere getirilir Enter tuşuna basılır. |
| 8. Single space | Satır aralarının boşluksuz olmasını sağlar. |
| 9. Double space | Satır aralarında boşluk verir. İki aralıklı yazımı sağlar. |
| A. Save | Bloklanan bölümü yeni bir isimle ayrı bir dosya olarak disk/diskete kaydeder. |
| B. Print | Bloklanan alanın yazıcıdan çıkmasını sağlar. |
| C. Calculate | Bloklanan alandaki sayısal değerler için hesapları yapar. |

Bloklamamanın aşamaları:

1. İmleç bloklanacak yerin başına getirilir.
2. **Ctrl+T** tuşlarına basılır.
3. Ok tuşları yardımı ile bloklanacak yerin sonuna kadar parlak şekilde işaretlenir.
4. **F10** tuşuna basılarak blok için yapılması istenen seçilir ve Enter tuşuna basılır.

5. SÜTUN BLOKLAMA

Mark rectangle (Ctrl-R)

PW ile sütun bloklama yapılabilir. Bu tip bloklama yapmak için **5. mark rectangle** seçeneği seçilerek ya da **Ctrl-R** tuşlarına basarak sütun bloklaması yapılabilir. Blok başı işaretinden sonra ok tuşları ile sağa sütun boyunca gidin ve sütun sonunda aşağı ok tuşu ile istenen yere kadar gelin. Artık bundan sonra normal bloklama ile yaptığımız işlemleri, blokladığımız sütun için de yapabilirsiniz. Aşağıdaki ekranda sütun bloklamasına örnek verilmiştir:

Not: Bu seçenek PW'nin eski versiyonunda yoktur.

8. ALT ÇİZGİLİ YAZDIRMA

Underline word (Ctrl-U)

İstenilen sözcük ya da sözcük gruplarının yazıcıdan altı çizili çıkması sağlanır. Kullanılışı koyu yazmada olduğu gibi altı çizilecek sözcüğün üzerine getirilerek **Ctrl-U** tuşlarına basılarak gerçekleştirilir.

Koyu yazma ve alt çizme işleminin ikisini birlikte aynı sözcüğe uygulayarak, hem koyu hem de altı çizili olmasını sağlayabilirsiniz.

9. ÇİZGİ ÇİZME

Draw lines (Ctrl-Y)

Yazım modundan çizgi moduna geçmeyi sağlar. Çizim modu ile dökümanınızda çizim yaparak, tablolarınızı çerçeve içine alarak daha güzel bir görünüm sağlayabilirsiniz.

Çizgi çizme moduna geçmek için **F3** menüsünden **9. Draw Lines** seçeneğini seçin. Yeni bir pencere açılarak karşımıza iki seçenek çıkacaktır.

Drawing	
▶ 1. Turn drawing on/off	Ctrl-Y
2. Select drawing style	
Esc-Cancel	← Select

1. Turn drawing on/off Ctrl-Y

Bu seçenek, daha önce seçilmiş bulunan çizim karakteri ile çizim yapmanızı sağlar.

2. Select drawing style

Bu seçeneğin seçilmesinden sonra karşımıza 4 yeni seçenek daha çıkacaktır. Buradan hangi çizim karakterini seçeceksek o çizim karakteri ile çizim yapmamızı sağlar.

Drawing Styles	
▶ 1. ———	
2. = = =	
3. ■■■■	
4. Other	
Esc-Cancel	← Select

Burada bulunan 4. seçenek olan **Other** seçeneği, diğer üç çizim karakteri dışında bizim belirleyebileceğimiz bir karakter ile çizim yapılması sağlanır.

Selection: -
Esc-Cancel ← Use selection

Selection: iletisinin karşısına bilgisayarda bulunan karakterlerden birini yazarak artık bu karakterle çizim yapabilirsiniz.

Örnek verecek olursak * işaretini yazarsanız, * ile çizim yaparsınız.

Çizgi çizme modunun kolay tuşu **Ctrl-Y** tuşlarıdır. Bu tuşlara ilk başta çizgi çizme moduna geçilir. Çizime ara verilmek istendiğinde tekrar **Ctrl-Y** tuşlarına basarak yazım moduna geçilebilir.

10. BUL ve DEĞİŞTİR

A. Find & Replace (Ctrl-F)

Döküman içerisinde bir sözcüğün aranıp bulunmasını ya da başka bir sözcükle değiştirilmesini sağlar.

Ctrl-F tuşlarına bastığımızda karşımıza yeni bir pencere gelecektir, burada gerekli verileri girerek işleminizi yapabilirsiniz.

Find and Replace	
Find:	
Ignore case (Y/N):	
Replace with:	
Manual or automatic (M/A):	
Esc-Cancel	← Continue

Find and Replace penceresinde bulunan giriş yerlerinin kullanımları şöyledir:

Find: Aranacak sözcük yazılmalıdır.

Replace with: Aranan sözcüğün yerine gelecek sözcük yazılmalıdır. (Bu bölüm boş geçilirse değiştirme yapılmaya yerine yalnızca sözcük bulma sağlanır.)

Ignore case (Y/N): Büyük, küçük harf ayırımı yapılıp yapılmayacağı belirtilir.

Manual or automatic (M/A): Buraya **M** ya da **A** harfleri yazılır. **M** yazıldığında sözcükler bulunarak size bulunan sözcüğün değiştirilip değiştirilmeyeceği sorulacaktır. **A** seçiminde ise size sorulmadan döküman içerisinde bulunan bütün sözcükler değiştirilecektir.

Yalnızca arama yapıldığında **manual** seçimde, aranan sözcüğün üzerinde durur. **Automatic** seçiminde ise aranılan karakter dizisinin döküman içerisinde kaç kez tekrar edildiği yazılır.

Değiştirme işleminde manual seçiminde aranan karakter dizisi bulunduğu pencere açılarak değiştirme yapılıp yapılmayacağı sorulur. Automatic seçiminde ise kaç sözcüğün değiştirildiği sayısal olarak verilir.

Aranan karakter grubu döküman içerisinde bulunamazsa bir mesajla iletir.

Manual seçimi yapıldığında aranan sözcük bulunur, ekranda yeni bir pencere belirir. Burada karşımıza iki seçenek çıkar:

▶ 1. Replace word and continue 2. Continue without replacing	
Esc-Cancel	← Select

1. Replace word and continue : Bulunan sözcük değiştirilir ve aramaya devam edilir.

2. Continue without replacing : Bulunan sözcük değiştirilmeden geçilir ve arama sürer.

11. HESAP YAPMA

Calcute (Ctrl-M)

Hesaplama işlerinde kullanılır, kolay tuşu **Ctrl-M**'dir.

Calculator			
Calculation:			
Result:			
Esc-Cancel	F8-Average	F9-Calculate	← Paste result

Gelecek pencerede **Calculation:** iletisinin karşısına rakam ve işlemler yazılır. **F9** tuşu ile sonuç hesaplatılır. **Enter** tuşuna bastığınızda işlemin sonucu döküman içinde imlecin bulunduğu yere yazılacaktır. **F8** sonucu yuvarlayarak verir.

Not: Bu bölüm PW'nin eski versiyonlarında yoktur.

J. BİÇİMLEME MENÜSÜ

F4-Format

F4 alt menüsü düzen, şekil menüsüdür. Burada sayfa düzenleme komutları bulunur. Bunlar: Sol-sağ marj ayarları, tab ayarı, kağıda yazım sırasında her sayfada çıkacak başlık ve dipnotların düzenlenmesi, sola dayalı yazım, sağa dayalı yazım, satıra ortalama gibi düzenleme komutlarıdır.

F4 tuşuna basıldığında karşımıza şu menü gelir:



1. SOL/SAĞ MARJ AYARI

Set left/right margins

Yazım alanında soldan ve sağdan bırakılacak boşluk sayısı ayarlanır.

Bu seçimi yaptığımızda karşımıza iki yeni seçenek çıkar:

Set Left and Right Margins	
▶ 1. Insert new margins	
2. Change margins of the current block	Ctrl-[
Esc-Cancel	←] Select

1. Insert new margins: Marjları yeniden ayarlamak içindir. Seçim sonrası sol ve sağ marjı belirlememiz için aşağıdaki pencere çıkar:

Margin Settings	
Left margin :	
Right margin:	
Esc-Cancel	←] Continue

Left margin : 10
Right margin : 70 şeklinde giriş yapılır.

Left margin'in karşısına girilecek sayı, sol taraftan kaç karakter boşluk bırakılacağını belirtir.

Right margin'in karşısına girilecek sayı, sağ tarafın kaçınıcı sütunda biteceğini belirtir.

2. Change margins of the current block Ctrl-[: Bloklanan bir alanın marjlarını değiştirir. Kolay tuşu **Ctrl-[** tuşlarıdır.

2. ÜST/ALT BOŞLUK & SAYFA SATIR AYARI

Set top/bottom margins & length

Üstten ve alttan kaç satır boşluk bırakılacağı ve bir sayfaya kaç satır yazı yazılacağı ayarlanır.

Bu seçimden sonra yeni bir pencere açılır:

Top/Bottom Margins and Page Length	
Top margin :	
Bottom margin:	
Page length :	
Esc-Cancel	↵ Continue

Top margin: Üst satırda kaç satır boşluk bırakılacağı belirtilir.

Bottom margin: Alt sırda kaç satır boşluk bırakılacağı belirtilir.

Page length: Bir sayfaya kaç satır yazılacağı belirtilir.

(PW'nin eski uyarlamalarında Sol/sağ marj ayarları ile Üst/alt boşluk ayarları aynı seçenek içerisindedir.)

3. DURAK YAPMA

Set tabs

Döküman içerisinde yapılacak tab (durak) ayarları burada yapılır. Bir çizelge hazırlarken bu duraklardan yararlanarak, çizelgeyi daha çabuk ve düzenli hazırlayabiliriz.

Bu seçimden sonra aşağıdaki pencere çıkar:

Set Tabs	
► 1. Insert new tabs	
2. Change tabs of the current block	Ctrl-K
Esc-Cancel	← Select

1. Insert new tabs: Tabları yeniden ayarlamak içindir.

2. Change tabs of the current block Ctrl-K : Bloklanan bir alanın tablarını değiştirir. Kolay tuşu **Ctrl-K**'dir.

Bu seçimlerin ikisinden sonra ekranın altında şu yazı çıkar:

Type T or D at each tab location

Sol/sağ ok tuşları ile imleç istenilen yere hareket ettirilir. Tab noktası belirlendikten sonra **T** ya da **D** tuşlarına basılarak tab yapılmış olur. Enter tuşuna basılarak tab ayarından çıkılır. Artık klavyede bulunan tab tuşuna her basışta bir sonraki tab noktasına, **Shift** ile birlikte **tab** tuşuna basılırsa bir önceki tab noktasına geçilir.

Eski tab noktalarını silmek için ise; tab ayarı seçilir, sol/sağ ok tuşları ile silinmek istenen tab noktasının üzerine gidilir ve aralık çubuğuna basılır.

Sayısal işlemler için **D** ile tab yapılabilir. **D** ile tab yapıldığında binler hanesini (,) virgül ile ayırır. Kuruşları da nokta ile ayırır. Sayısal tab olduğu için sayıları sağa dayalı yazar.

123,456.00
12,345.50
1,234,567.00

 şeklinde yazar.

Sütun bloklamada verdiğimiz ekrana bakarsanız, sayılar **D** ile tab yapılarak yazılmıştır.

4. SAYFA ÜZERİ, BAŞLIK YAZIMI

Set header

Kağıda döküm yapıldığında sayfaların üst kısmında çıkması istenen açıklamalar ve sayfa numarası vermek için kullanılır.

Header		
Line 1:		
Line 2:		
Position (Center/Left/Right):		
Style (Normal, Boldface, Underline, Italics):		
Left margin:		
Right margin:		
First page:		
Font: Courier P 12		
Esc-Cancel	F8-Change font	← Continue

Burada ilk iki satıra yazılması istenen açıklamalar yazılabilir. Eğer dökümanımıza sayfa numarası vereceksek kaçınıcı satırda olmasını istiyorsak o satıra gelip * işareti içerisine rakamı yazmamız gerekir. *1* gibi... diğer sayfaları PW kendisi numaralandıracaktır.

Line 1: Birinci satıra yazılması istenen açıklama ya da sayfa numarası.

Line 2: İkinci satıra yazılması istenen açıklama ya da sayfa numarası.

Position (Center/Left/Right): Açıklamamanın satırın neresine yazılması gerektiği. **Center** ise **C** seçilir ortaya yazar, **Left** ise **L** seçilir sola yazar, **Right** ise **R** seçilir sağ tarafa yazar.

Style (Normal, Boldface, Underline, Italics): Açıklama ve sayfa numarasının yazı stili belirtilir. **N** normal yazı, **B** koyu yazı, **U** alt çizgili yazı, **I** italik yazı şeklinde yazılmasını sağlar. Hem koyu, hem alt çizgili yazdırmak için **B,U** şeklinde yazabiliriz.

Left margin: Solda kalacak boşluk.

Right margin: Sağ tarafın kaçınıcı sütunda biteceği.

First Page: Kaçınıcı sayfadan başlayarak yazılacağını belirtir.

F8 Change Font: Bu bölümün hangi fontta yazılacağı seçilir.

5. SAYFA ALTI, DİPNOT YAZIMI

Set footer

Kağıda döküm yapıldığında sayfaların alt kısmında çıkması istenen açıklamalar ve sayfa numarası için kullanılır.

Footer		
Line 1:		
Line 2:		
Position (Center/Left/Right):		
Style (Normal, Boldface, Underline, Italics):		
Left margin:		
Right margin:		
First page:		
Font: Courier P 12		
Esc-Cancel	FB-Change font	← Continue

Burada sayfanın altındaki ilk iki satıra yazılması istenen açıklamalar yazılabilir.

Karşımıza gelen pencere ile **Set header**'da yaptığımız gibi aynı seçimleri bu kez sayfanın altında çıkacağını düşünerek yapabiliriz.

6. SATIR BAŞINI GEÇİCİ OLARAK DEĞİŞTİRME

Turn indent on/off (Ctrl-N)

Yazım sırasında satır başının geçici olarak değiştirilmesi istendiğinde buradan seçim yapılır ya da kolay tuşu **Ctrl-N** kullanılır. Satır başı yapılacak yere imleç getirilir, **Ctrl-N** tuşlarına basılır. Altındaki sütun cetvelinde > işareti çıkar. Satır başı artık geçici olarak buraya alınmıştır. Tekrar **Ctrl-N** tuşlarına basarak satır başını kaldırabiliriz.

7. SATIR ARALIĞINI İKİ YA DA TEK YAPMA

Turn double spacing on/of (Ctrl-D)

Satır aralığı normal yazımda **bir**'dir. Satır aralarının çift olmasını istediğimiz zaman buradan seçim yaparız ya da kolay tuşları **Ctrl-D** ile ayarlayabiliriz.

Ctrl-D tuşlarına ilk basışta iki ara, ikinci basışta yine normal ara yazıma geçer.

8. YAZIYI SATIRA ORTALAMA

Center line (Ctrl-X)

Yazım sırasında, yazılan başlık vb. yazıları satıra ortalar. Başlık olarak belirlediğiniz bir satıra imleci getirin ve **Ctrl-X** tuşlarına basın. Yazının satıra ortalandığını göreceksiniz.

9. SATIRI SOLA YASLAMA

Left justify line

Döküman içinde istenilen satırı sola dayalı olarak yazdırır. İmleci istediğiniz bir satıra getirip **F4** menüsünden **9. Left justify line** seçeneğini seçtiğinizde satırın sola dayalı yazıldığını göreceksiniz.

10 SATIRI SAĞA YASLAMA

A. Right justify line

İstenilen satırların sağa dayalı olarak yazılması sağlanır. İmleci istediğiniz bir satıra getirip **F4** menüsünden **A. Right justify line** seçeneğini seçtiğinizde satırın sağa dayalı yazıldığını göreceksiniz.

```
F1-Help F2-File/Print F3-Edit F4-Format F5-Dictionary F6-Addresses
ORTALANMIŞ YAZI
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

SOLA DAYANMIŞ YAZI
XXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

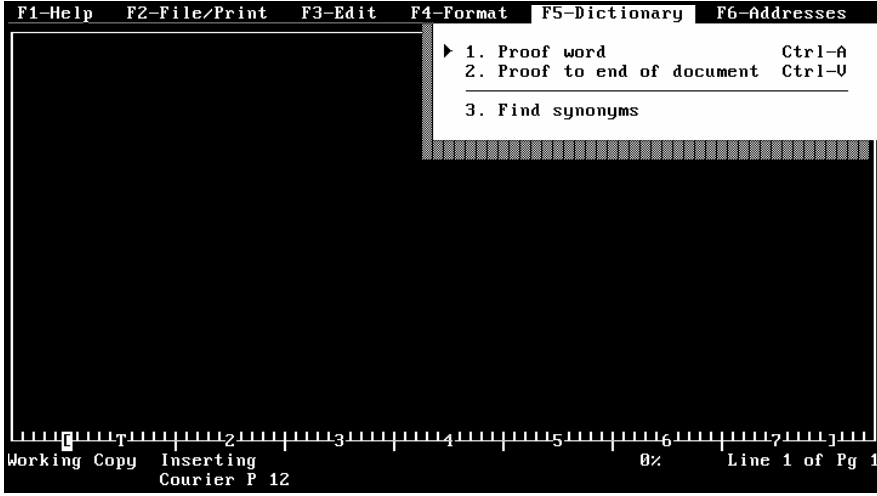
SAĞA DAYANMIŞ YAZI
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

Working Copy Inserting 0% Line 1 of Pg 1
Esc-Main Menu Courier P 12
```

K. YAZIM KLAVUZU

F5-Dictionary

Bu bölümde yazım içinde yanlış yazımların **PW** tarafından bulunarak düzeltilmesi, sözcüklerin ve eş anlamlı sözcüklerin bulunması sağlanır. **PW**'nin kendi içinde barındırdığı sözlük İngilizce olduğu için, İngilizce'ye göre tarama yapacaktır.

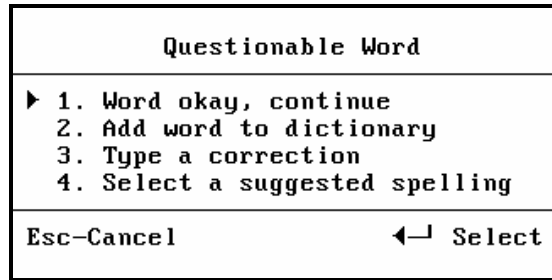


1. Sözcük Kontrolü

Proof word (Ctrl-A)

İmlecın üzerinde durduđu sözcüğü kontrol eder. Sözlüğünden bakarak olup, olmadıđı ya da yanlış olup olmadıđını kontrol eder.

Seçim sonucu şu pencere gelir:



1. **Word okay, continue:** Sözcükleri değiştirmeden devam eder.
2. **Add word to dictionary:** İşaretlenen sözcüğü, sözlüğüne ekler.
3. **Type a correction:** Sözcüğü verilen sözcük ile değiştirir.
4. **Select a suggested spelling:** Sözcüğün doğru yazılışını gösterir.

2. Yanlış Sözcükleri Bulma

Proof document (Ctrl-V)

İmlecin bulunduğu yerden başlayarak tüm dökümanı arar ve yanlış sözcükleri bulur. Bu bölümün seçimi sonrasında **1. Proof word** menüsü altında çıkan seçenekler burada da çıkar. Kullanımı aynıdır.

3. Eşanlamlıları Bulma

Find synonyms

Sözcüğün eşanlamlılarını göstererek, bunlardan birini seçmemizi sağlar.

L. ADRES MENÜSÜ

F6/Addresses

Bu menü ile adres dosyasına girerek, yeni adres girme, kayıt düzeltme, kayıt arama ve eski adreslerin kullanımını sağlar.



1. Select address book: Bu seçenekle istenilen adres dosyası seçimi sağlanır ya da F7 ile yeni bir adres dosyası oluşturulur.

Name of book to use:		

Esc-Cancel	F7-Create new book	← Use existing book

2. Add an address: Adres dosyasına yeni adreslerin girilmesi yapılır. F10 tuşu ile disk/diskete kayıt edilir.

Address Book	
Title :	
First name:	
Last name :	
Position :	
Company:	
Address:	
:	
City :	Zip:
State:	
Work phone:	Home phone:
Comment 1 :	
Comment 2 :	
Enter new information.	
Esc-Cancel	F10-Add

Adres menüsündeki terimlerin anlamları:

Title	: Ünvan		
First name	: İlk Adı		
Last name	: Soyadı		
Position	: Pozisyonu		
Company	: Şirket		
Address	: Adres		
City	: Şehri		
State	: Ülke	Zip	: Posta Kodu
Work phone	: İş Telefonu	Home phone	: Ev Telefonu
Comment	: Düşünceler, not		

3. Find an address: İstenilen bir adresi bulmak, bu adres üzerinde düzeltme yapmak, silmek için kullanılır.

Address Book	
Title :	
First name:	
Last name :	
Position :	
Company:	
Address:	
City :	
State:	Zip:
Work phone:	Home phone:
Comment 1 :	
Comment 2 :	
F8-Copy to working copy Esc-Cancel	F9-Delete F10-Continue

- F8** PW içinde üzerinde çalıştığımız dökümana, seçilen adresi kopyalar.
F9 Seçilen adresi siler.
F10 Bir sonraki kaydı arar.

4. Specify copy format: Adresin dökümana yazım şekli belirlenir.

Address Book	
Line 1: *Title* *First name* *Last name*	
Line 2: *Position*	
Line 3: *Company*	
Line 4: *Address*	
Line 5: *City*, *State* *Zip*	
Line 6:	
Line 7:	
Line 8:	
Line 9:	
Esc-Cancel	Enter new copy format. F10-Save

Bu bölümde adres zarfının ya da dökümünün nasıl olacağı belirtilir. Yukarıdaki örneği dikkate alacak olursak, çıktı aşağıdaki gibi olur:

Ünvan İlk Adı Soyadı
Pozisyonu
Şirket
Adres
Şehir Ülke Posta Kodu

Örnek:

Öğr.Gör. Güven Tanış

Akdeniz Üniversitesi
Sosyal Bilimler Meslek Yüksekokulu
Antalya Türkiye 07150

Şeklinde olacaktır.

Kolay Tuşları Listesi:

Tuş	Görevi	Geçtiği Sayfa
Ctrl-A	Sözcük Kontrolü.	230
Ctrl-B	Sözcük koyulaştırma.	219
Ctrl-C	Bloklanan alanı kopyalama.	217
Ctrl-D	Çift ara, tek ara yapma.	228
Ctrl-F	Sözcük bulma, değiştirme.	221
Ctrl-G	Disketten dosya çağırma.	200
Ctrl-I	Satır aralama.	215
Ctrl-K	Bloklanan alanın tablalarını değiştirme.	226
Ctrl-L	Satır silme.	215
Ctrl-M	Hesap yapma.	222
Ctrl-N	Satır başını geçici olarak değiştirme.	228
Ctrl-O	Yazıcıya yazdırma.	205
Ctrl-P	Blok yapıştırma.	219
Ctrl-R	Sütun bloklama.	218
Ctrl-S	Dosyayı diske kayıt etme.	202
Ctrl-T	Blok yapma.	215
Ctrl-U	Sözcük altı çizdirme.	220
Ctrl-V	Yanlış sözcükleri bulma.	231
Ctrl-W	Sözcük silme.	315
Ctrl-X	Yazıyı satıra ortalama.	229
Ctrl-Y	Çizim moduna geçme.	220
Ctrl-]	Bloklanan alanın marjlarını değiştirme.	218, 224
Ctrl-PrtSc	Baskı önizleme.	207
Alt-0	Makro yapma.	211

SORULAR

1. Kelime İşlem (Editör) nedir?
2. Bloklama ve yapıştırma nedir?
3. Kelime işlem programlarında format ne anlama gelir?
4. Setup Menü'nün görevi nedir?
5. Makro yapma nedir? Hangi durumlarda yapılması gerekir?
6. Bir paragrafın bloklayıp kesilerek, başka bir yere taşınmasını aşama aşama anlatınız.
7. Makro yapmayı aşama aşama anlatınız.
8. Aşağıdaki ikili tuş kullanımlarının görevleri nelerdir?
 - a. Ctrl-C
 - b. Ctrl-B
 - c. Ctrl-U
 - d. Ctrl-X
 - e. Ctrl-L
 - f. Ctrl-I
 - g. Ctrl-T
 - h. Ctrl-G
 - i. Ctrl-S
 - l. Ctrl-R
 - k. Ctrl-W
 - m. Alt-0

BÖLÜM VII. SAYI SİSTEMLERİ

Nesnelerin miktarını belirleyen sembollere sayı denir. Bu sayıların bir araya getirilerek dizi oluşturulmasına da **sayı sistemi** denir. Biz günlük yaşantımızda bu türden işlerimizi onluk sisteme göre yapıyoruz. Yani **0,1,2,3,4,5,6,7,8,9** sembollerini kullanıyoruz. Bilgisayarlar da ikili, sekizli, onaltılık sayı sistemi ile çalışırlar.

Bilgisayar, programı derledikten sonra, kendi anlayacağı dil olan sisteme çevirir. Bu işi bilgisayar, kendiliğinden yapar. Bilgisayarlar, daha çok ikili sayı sistemini kullanırlar.

A. İKİLİ SAYI SİSTEMİ (BINARY NUMBER SYSTEM)

Bu sistemdeki semboller **0** ve **1** sayılarından oluşur. Tabanı **2**'dir. Yani basamaklar ikinin kuvvetlerinden oluşur.

Onluk sistemde sayıların basamakları **10** tabanına göre oluşur. **3597** sayısını buna göre inceleyelim:

$$\begin{array}{rcccccc} 3 & + & 5 & + & 9 & + & 7 & \text{kuvvetleri} \\ 10^3 & + & 10^2 & + & 10^1 & + & 10^0 & \text{kuvvetleri} \\ (1000 \cdot 3) & + & (100 \cdot 5) & + & (10 \cdot 9) & + & (1 \cdot 7) & \text{karşılıkları} \\ 3000 & + & 500 & + & 90 & + & 7 & = 3597 \text{ sonuç} \end{array}$$

Ya da, şu şekilde de açıklayabiliriz:

$$\begin{array}{l} \begin{array}{c} 3 \quad 5 \quad 9 \quad 7 \\ \left. \begin{array}{l} \rightarrow \\ \rightarrow \\ \rightarrow \\ \rightarrow \end{array} \right\} \end{array} \\ \begin{array}{l} \rightarrow \text{ Birler Basamağı} \quad 10^0 \quad 7 \times 1 \quad = \quad 7 \\ \rightarrow \text{ Onlar Basamağı} \quad 10^1 \quad 9 \times 10 \quad = \quad 90 \\ \rightarrow \text{ Yüzler Basamağı} \quad 10^2 \quad 5 \times 100 \quad = \quad 500 \\ \rightarrow \text{ Binler Basamağı} \quad 10^3 \quad 3 \times 1000 \quad = \quad +3000 \end{array} \\ \hline 3597 \end{array}$$

İkili sistemde de **2** tabanına göre oluşurlar:

$$2^0, 2^1, 2^2, 2^3, \dots, 2^n \text{ gibi...}$$

İkili sayı sisteminde, basamaklar şu şekilde olur: (1101 İkili Sayısı)

$$\begin{array}{ccccccccccc}
 1 & & 1 & & 0 & & 1 & & & & \\
 2^3 & + & 2^2 & + & 2^1 & + & 2^0 & & & & \\
 (8*1) & + & (4*1) & + & (2*0) & + & (1*1) & & & & \\
 & & & & & &) & & & & \\
 8 & + & 4 & + & 0 & + & 1 & & & & =13
 \end{array}$$

Bu işlemi açacak olursak;

1	1	0	1				
					→	Birler Basamağı	2^0
					→	İkiler Basamağı	2^1
					→	Dörtler Basamağı	2^2
					→	Sekizler Basamağı	2^3

$1 \times 1 = 1$	$0 \times 2 = 0$	$1 \times 4 = 4$	$1 \times 8 = 8$	+	8
					13

Not: 0 (Sıfır) dışındaki herhangi bir sayının 0'ıncı kuvveti 1'dir.

Bu işlemi şu biçimde de yapmak olanaklıdır:

Basamakların karşısına gelen sayıları çarparak toplayabiliriz. Ya da karşısına 1 rakamı gelenleri var, 0 rakamı gelenleri yok sayarak eldekileri toplayarak da buluruz.

$$\begin{array}{ccccccccccc}
 1 & & 1 & & 0 & & 1 & & & & \\
 8'ler\ bas. & & 4'ler\ bas. & & 2'ler\ bas. & & 1'ler\ bas. & & & & \\
 8*1 & + & 4*1 & + & 2*0 & + & 1*1 & & & & \\
 8 & + & 4 & + & 0 & + & 1 & & & & =13
 \end{array}$$

Şimdi, yukarıda yaptığımız örnekleri inceleyerek, 10'luk ve 2'lik sistemdeki benzerlikleri görünüz ve hangi yol size kolay geliyorsa o yolu izleyiniz.

1. İKİLİ SAYI SİSTEMİNDE DÖRT İŞLEM

TOPLAMA

İkili sayı sisteminde toplama yaparken, aşağıdaki kurallar geçerlidir.

$$\begin{array}{l}
 0 + 0 = 0 \\
 0 + 1 = 0 \\
 1 + 0 = 1 \\
 1 + 1 = 0 \rightarrow \text{elde 1 kalır}
 \end{array}$$

Onluk sistemde yaptığımız gibi toplam $1+1=2$ ise ikili sayı sisteminde 2 olmadığı için 0 yazılıp elde kalan 1 ikinci haneye eklenir.

<u>Onluk</u>	<u>İkili</u>	<u>Onluk</u>	<u>İkili</u>
8	1000	11	1011
+ 6	0110	+ 7	+ 0111
14	1110	18	10010
24	11000	70	1000110
+ 61	111101	+ 33	+ 100001
85	1010101	103	1100111

Ç I K A R M A

Çıkarma işleminde aşağıdaki kurallara uyulur:

0	-	0	=	0	
1	-	1	=	0	
1	-	0	=	1	
0	-	1	=	1	→ Sol basamaktan bir ödünç almak koşuluyla 1 kalır. Sol basamaktan alınan 1 sayısı sağa 2 olarak geçer.

<u>Onluk</u>	<u>İkili</u>	<u>Onluk</u>	<u>İkili</u>
12	1100	72	1001000
- 7	- 111	- 27	- 11011
5	0101	45	101101
8	1000	135	10000111
- 3	- 11	- 86	- 1010110
5	0101	49	00110001

Ç A R P M A

İkili sistemde çarpma işlemi aşağıdaki şekilde yapılır:

1	*	1	=	1
1	*	0	=	0
0	*	0	=	0
0	*	1	=	0

<u>Onluk</u> 7 <u>x 4</u> 28	<u>İkilik</u> 111 <u>x 100</u> 000 000 + 111 11100 = 28	<u>Onluk</u> 24 <u>x 5</u> 120	<u>İkilik</u> 11000 <u>x 101</u> 11000 00000 + 11000 1111000 = 120
---------------------------------------	---	---	--

<u>Onluk</u> 26 <u>x 7</u> 182	<u>İkilik</u> 11010 <u>x 111</u> 11010 11010 + 11010 10110110	<u>Onluk</u> 21 <u>x 14</u> 84 + 21 294	<u>İkilik</u> 10101 <u>x 11110</u> 00000 10101 10101 + 10101 100100110
---	---	--	---

BÖLME

Bölme işlemini 10'luk sistemdeki gibi yaparız.

<u>Onluk</u> 57 5 <u>-5</u> 11 07 <u>-5</u> 2	<u>İkilik</u> 111001 101 <u>-101</u> 1011 01000 <u>- 101</u> 0111 <u>- 101</u> 010 = 2	<u>Onluk</u> 26 2 <u>-2</u> 13 06 <u>-6</u> 0	<u>İkilik</u> 11010 10 <u>-10</u> 1101 010 <u>- 10</u> 0010 <u>- 10</u> 00
--	---	--	---

<u>Onluk</u> 28 7 <u>-28</u> 4 00	<u>İkilik</u> 11100 111 <u>-111</u> 100 000	<u>Onluk</u> 36 6 <u>-36</u> 6 00	<u>İkilik</u> 100100 110 <u>-110</u> 110 00110 <u>- 110</u> 000
--	--	--	--

2. İKİLİ SİSTEMDEN ONLUK SİSTEME GEÇİŞ

Daha önce **10**'luk sistemde basamakların **10**'un kuvvetleri, **2**'lik sistemde basamakların **2**'nin kuvvetleri olduğunu söylemiştik. Buna göre, **2**'lik sistemde yazılan bir sayı, **10**'luk sisteme şöyle çevrilir:

101001011 ikili sayısını onluk sayıya çevirelim:

$$\begin{array}{cccccccc} 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 2^8 & + & 2^7 & + & 2^6 & + & 2^5 & + & 2^4 & + & 2^3 & + & 2^2 & + & 2^1 & + & 2^0 & \text{Kuvveti} \\ 256 & + & 128 & + & 64 & + & 32 & + & 16 & + & 8 & + & 4 & + & 2 & + & 1 & \text{Karşılığı} \end{array}$$

Bu kuvvetlerin karşılığına gelen sayılarla çarpımı toplanarak **10**'luk sistemdeki karşılığı bulunur.

$$\begin{array}{cccccccccccc} 256 \times 1 & + & 128 \times 0 & + & 64 \times 1 & + & 32 \times 0 & + & 16 \times 0 & + & 8 \times 1 & + & 4 \times 0 & + & 2 \times 1 & + & 1 \times 1 \\ 256 & + & 0 & + & 64 & + & 0 & + & 0 & + & 8 & + & 0 & + & 2 & + & 1 & = & 331 \end{array}$$

Ya da;

1	0	1	1	0	1	
						1'ler basamağı
						2 " "
						4 " "
						8 " "
						16 " "
						32 " "
						1x1 = 1
						2x0 = 0
						4x1 = 4
						8x1 = 8
						16x0 = 0
						32x1 = +32
						45

Buraya kadar öğrendiklerimizden şunu çıkartabiliriz: Kuvvet karşılıkları, karşısına gelen sayılarla çarpılıyordu. Bu sistemde **0**'la çarpma da olduğundan, bu işlemleri daha kısa yoldan yapabiliriz. **0**'la çarpılacak sayıları çizerek kalanları topladığımızda aynı sonucu daha kısa yoldan elde etmiş oluruz.

$$\begin{array}{cccccccc} 1 & 0 & 1 & 1 & 0 & 1 \\ 32 \times 1 & + & 16 \times 0 & + & 8 \times 1 & + & 4 \times 1 & + & 2 \times 0 & + & 1 \times 1 \\ 32 & + & 0 & + & 8 & + & 4 & + & 0 & + & 1 & = & 45 \end{array}$$

3. ONLUK SİSTEMDEN İKİLİ SİSTEME GEÇİŞ

Onluk sistemde yazılan bir sayıyı ikili sayıya çevirmek için, sayı sürekli 2'ye bölünür. Burada dikkat edeceğimiz nokta, çıkan sonuçların tam olmasıdır.

$$\begin{array}{r}
 162 \quad | \quad 2 \\
 -162 \quad | \quad 81 \quad | \quad 2 \\
 \hline
 \textcircled{0} \quad -80 \quad | \quad 40 \quad | \quad 2 \\
 \hline
 \textcircled{1} \quad -40 \quad | \quad 20 \quad | \quad 2 \\
 \hline
 \textcircled{0} \quad -20 \quad | \quad 10 \quad | \quad 2 \\
 \hline
 \textcircled{0} \quad -10 \quad | \quad 5 \quad | \quad 2 \\
 \hline
 \textcircled{0} \quad -4 \quad | \quad 2 \quad | \quad 2 \\
 \hline
 \textcircled{1} \quad -2 \quad | \quad 1 \\
 \hline
 \textcircled{0}
 \end{array}$$

Bu işlemden sonra kalanlar, sağdan itibaren teker teker alınarak, soldan sağa doğru yanyana yazılır.

$$(162)_{10} = (10100010)_2 \text{ sonucunu buluruz.}$$

Onluk 162 sayısının, ikili sistemdeki karşılığı **10100010** sayıdır.

$$\begin{array}{r}
 127 \quad | \quad 2 \\
 -126 \quad | \quad 63 \quad | \quad 2 \\
 \hline
 \textcircled{1} \quad -62 \quad | \quad 31 \quad | \quad 2 \\
 \hline
 \textcircled{1} \quad -30 \quad | \quad 15 \quad | \quad 2 \\
 \hline
 \textcircled{1} \quad -14 \quad | \quad 7 \quad | \quad 2 \\
 \hline
 \textcircled{1} \quad -6 \quad | \quad 3 \quad | \quad 2 \\
 \hline
 \textcircled{1} \quad -2 \quad | \quad 1 \\
 \hline
 \textcircled{1}
 \end{array}$$

Çıkan sonucu yine sağdan sola doğru yanyana yazdığımızda, **1111111** ikili sayısını buluruz.

Bu işlemi şu şekilde de yapabiliriz:

$$\begin{array}{r|l}
 121 & 1 \\
 60 & 0 \\
 30 & 0 \\
 15 & 1 \\
 7 & 1 \\
 3 & 1 \\
 1 & 1
 \end{array}$$

Sayıları alt alta inerken sürekli yine ikiye böleriz. Bu bölümlerde sayılar tekli ise kalan 1 sağ tarafa yazılır. Kalan yoksa 0 sayısı yine yazılır. Sonuçta sayılar bu kez alttan başlayarak, yukarıya doğru yazılınca ikili karşılığı bulunur.

1111001 sayısı **121** sayısının 2'lik karşılığıdır.

67	1	↑	67 tek sayı olduğu için 1 yazılıp 66 sayısı 2'ye bölünür. 16 çift olduğundan 0 yazılır.
33	1		
16	0		
8	0		
4	0		
2	0		
1	1	67 onluk sayısının ikili karşılığı 1000011'dir.	

B. ONALTILIK SAYI SİSTEMİ (HEXADECIMAL NUMBER SYSTEM)

İkili sistemde program yazmak oldukça güçtür. Bu türden sayıları girerken hata yapma olasılığı çoktur. Bu nedenle 16'lık sayı sistemi ile bilgisayara giriş yapmak daha kolay olduğundan bu sistem yeğlenir.

Onaltılık sistemde sayılar, 16'nın kuvvetlerinden oluşmuştur. Bu sistemde 0'dan 9'a kadar rakamların kullanılmasının yanısıra, İngiliz alfabesindeki ilk 6 harf kullanılmıştır. A, B, C, D, E ve F harfleri eklenmiştir.

Aşağıdaki tablo bu karşılıkları vermektedir:

Onluk Sayılar	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
Onaltılık Sayılar	0 1 2 3 4 5 6 7 8 9 A B C D E F

Onaltılık sistemde sayılar 16'nın kuvvetlerinden oluşur.

16^0 , 16^1 , 16^2 , 16^3 16^n

1. ONALTILIK SAYI SİSTEMİNDE DÖRT İŞLEM

TOPLAMA

Bu sistemde de, onluk ve ikili sistemde olduğu gibi toplama yapılır. Dikkat edeceğimiz nokta: Tabanın 16 olduğu ve toplanan sayılar 16 ve 16'nın katı ise elde var işleminin yapılacağıdır.

<u>Onluk</u> 172 + 219 <hr/> 391	<u>Onaltılık</u> AC + DB <hr/> 187	<u>Onluk</u> 940 + 123 <hr/> 1063	<u>Onaltılık</u> 3AC + 7B <hr/> 427
---	---	--	--

Sol tarafta yaptığımız toplamın açıklamasını yapalım: C(12) ve B(11) toplanır 23 elde edilir. 16 eldeye alınır, 7 kalır. İkinci basamaktaki A(10) ve D(13) toplanır, 23 yapar. Elde bir eklendiğinde 24 olur. 16 eldeye alınır. 8 kalır. Elde 1 sol başa yazılır ve 187 onaltılık sayısı bulunur.

<u>Onluk</u>	<u>Onaltılık</u>	<u>Onluk</u>	<u>Onaltılık</u>
3115	C2B	210	D2
+ 173	+ AD	+ 172	+ AC
<u>3288</u>	<u>CD8</u>	<u>382</u>	<u>17E</u>

Ç I K A R M A

Çıkarma işleminde de kurallarımız aynıdır. Unutmayacağımız şey basamakların 16 tabanına göre olduğudur.

<u>Onluk</u>	<u>Onaltılık</u>	<u>Onluk</u>	<u>Onaltılık</u>
2588	A1C	3245	CAD
- 188	- BC	- 2779	- ADB
<u>2400</u>	<u>960</u>	<u>466</u>	<u>1D2</u>

C(12)'den, C(12) çıkınca 0 (Sıfır) kalır. 1'den B(11) çıkmayacağından soldan bir 16'lık alınır. 16+1=17 olur. 17'den B(11) çıkınca 6 kalır. Solda A(10)'dan bir almıştık, geriye 9 kaldı. Bunu aşağıya alırsak. Sonuç 960 olur.

<u>Onluk</u>	<u>Onaltılık</u>	<u>Onluk</u>	<u>Onaltılık</u>
173	AD	223	DF
- 126	- 7E	- 162	- A2
<u>47</u>	<u>2F</u>	<u>61</u>	<u>3C</u>

Ç A R P M A

İkili sistemdeki gibi 16 ve 16'nın katları sol tarafa yazılır.

<u>Onluk</u>	<u>Onaltılık</u>	<u>Onluk</u>	<u>Onaltılık</u>
2861	B2D	701	2BD
x 11	x B	x 195	x C3
<u>31471</u>	<u>7AEF</u>	<u>136695</u>	<u>837</u>
			+ 20DC
			<u>215F7</u>

Solda yaptığımız örneğin açıklaması: B(11) ile D(13)'ün çarpımı 143 yapar. 143'te 8 kez vardır. 8 adet onaltılık sayıyı eldeye atarsak, F(15) kalır. 2 ile B(11)'i çarptığımızda 22 eder. Elde kalan 8'i de eklersek 30 eder. 30'da 16, 1 kez var. Bunu eldeye atarsak, geriye E(14) kalır. B(11) ile B(11)'yi çarparsak 121 yapar. Elde 1 kalmıştı bunu da eklersek 122 eder. 122'de 16, 7 kez var. Bunları atarsak geriye A(10) kalır. A'yı aşağıya yazarız. Elde 7 kalmıştı bunu da sol tarafa yazarız, sonuç 7AEF olur.

Bir örnek de sağlamasıyla birlikte yapalım:

$$\begin{array}{r}
 17F \longrightarrow 503 \\
 \times AA \longrightarrow \times 170 \\
 \hline
 13A6 \\
 + 13A6 \\
 \hline
 14E06
 \end{array}$$

$$\begin{array}{r}
 85510 \quad | \quad 16 \\
 - 85505 \quad | \quad 5344 \\
 \hline
 6 \quad | \quad -5344 \quad | \quad 16 \\
 \quad \quad | \quad 334 \quad | \quad 16 \\
 \quad \quad | \quad 20 \quad | \quad 16 \\
 \quad \quad | \quad 14 \quad | \quad -16 \quad | \quad 16 \\
 \quad \quad | \quad (E) \quad | \quad 4 \quad | \quad 1
 \end{array}$$

Sağdan sola doğru alırsak 14E06 onaltılık sayısı çıkar.

B Ö L M E

Bölme işleminde de aynı kuralları gözönünde bulundurmanız gerekmektedir.

<u>Onluk</u>	<u>Onaltılık</u>
$ \begin{array}{r} 3283 \quad \quad 11 \\ -22 \quad \quad 298 \\ \hline 108 \\ -99 \\ \hline 93 \\ -88 \\ \hline 5 \end{array} $	$ \begin{array}{r} CD3 \quad \quad B \\ -B \quad \quad 12A \quad (298) \\ \hline 1D \quad (29) \\ -16 \quad (22) \\ \hline 073 \quad (115) \\ -6E \quad (110) \\ \hline 5 \quad (5) \end{array} $

C (12)'de B(11), 1 kez var. 1*B=B yapar. C(12)'den B(11) çıkınca 1 kalır. Yukarıdan D(13)'yi aşağıya alırsız. 1D olur. 1D'de B(11), 2 kez var. 2*B=16 yapar. 1D'den 16 çıkınca 7 kalır. 3'ü aşağıya ineriz. 73(115) olur. 73'te B(11), A(10) kez var. B*A=6E yapar. 73-6E=5 kalır. Sonuç 12A'dır.

<u>Onluk</u>	<u>Onaltılık</u>
$ \begin{array}{r} 384676 \quad \quad 2754 \\ -2754 \quad \quad 139 \\ \hline 10927 \\ -8262 \\ \hline 26656 \\ -24786 \\ \hline 01870 \end{array} $	$ \begin{array}{r} 5DEA4 \quad \quad AC2 \\ -5610 \quad \quad 8B \quad (139) \\ \hline 07DA4 \quad (32164) \\ -7656 \quad (30394) \\ \hline 074E \quad (1870) \end{array} $
$ \begin{array}{r} 51787 \quad \quad 178 \\ -356 \quad \quad 290 \\ \hline 1618 \\ -1602 \\ \hline 167 \end{array} $	$ \begin{array}{r} CA4B \quad \quad B2 \\ -B2 \quad \quad 122 \quad (290) \\ \hline 184 \quad (388) \\ -164 \quad (356) \\ \hline 20B \quad (523) \\ -164 \quad (356) \\ \hline A7 \quad (167) \end{array} $

4. ONALTILIK SİSTEMDEN İKİLİ SİSTEME GEÇİŞ

Onaltılık sayı sisteminden ikili sayı sistemine geçmek için, 16'lık sayının her basamağı ayrı ayrı ikili sisteme çevrilir, sonra bunlar birleştirilir. İkilik düzene çevirirken her 16'lık sayının basamağı dört basamaklı ikili sayıdan oluşacağından, basamak sayısını 4'e çıkarmak için sol tarafa gereken kadar 0 eklenir. Ama unutmayalım ki bu 0'ların değeri yoktur.

7	C	E	16'lık sayı
7	12	14	Basamakların 10'luk karşılığı
0111	1100	1110	Basamakların 2'lik karşılığı

11111001110 ikili sayısı elde edilir.

2	A	6	D	4
2	10	6	13	4
0010	1010	0110	1101	0100

101010011011010100 ikili sayısını elde ederiz.

5. İKİLİ SİSTEMDEN ONALTILIK SİSTEME GEÇİŞ

İkili sayı sisteminden, onaltılık sayı sistemine geçerken; ikili sistemde yazılan sayı sağdan itibaren dörder ayrılır. Kalanlara sıfır eklenir. Bundan sonra elde edilenler onaltılık sayıya çevrilir.

101100111111101 ikili sayısını, onaltılık sayıya çevirelim.

<u>0101</u>	<u>1001</u>	<u>1111</u>	<u>1101</u>
5	9	15	13
		(F)	(D)

101100111111101 ikili sayısının onaltılık karşılığı **59FD**'dir.

10111011011011 ikili sayısını çevirelim:

<u>0010</u>	<u>1110</u>	<u>1101</u>	<u>1011</u>
2	14	13	11
	(E)	(D)	(B)

10111011011011 ikili sayısının onaltılık karşılığı **2EDB**'dir.

SORULAR

1. 7864 onluk sayısını ikili sayıya çeviriniz?
2. 1101101 ve 10110 ikili sayıların toplamını? Çıkan sonucu onluk sayıya çeviriniz.
3. 101110110 ikili sayısını onluk sayıya çeviriniz.
4. 101101111011 ikili sayısını onaltılık sayıya çeviriniz.
5. AC8E onaltılık sayısını onluk sayıya çeviriniz.
6. F8CA onaltılık sayısından A2 onaltılık sayısını çıkarınız.
7. 101101111 ikili sayısını onluk sayıya çeviriniz.
8. 1011100110 ikili sayısını onluk sayıya çeviriniz.
9. F3AD onaltılık sayısını onluk sayıya çeviriniz.
10. 1000100001011101 ikili sayısını onaltılık sayıya çeviriniz.

BÖLÜM VIII. PROGRAMLAMAYA GİRİŞ

A. OPERATÖRLER

1. Aritmetik İşlem Operatörleri

Aritmetik işlem operatörleri, günlük yaşamdaki işlem mantığı ile aynıdır. Yazılış biçimi ile bazı değişiklikler vardır. Bunlar şöyledir:

Cebirsel ve matematiksel ifadelerin bilgisayarda yazımı:

<u>Cebirsel İfade</u>	<u>Bilgisayar Yazımı</u>
$\frac{x-y}{z}$	$(x-y) / z$
$\frac{xy}{z}$	$x*y/z$
$\frac{x+y}{z}$	$(x+y) / z$
$(x)^{2^y}$	$(x^2)^y$
x^{y^z}	$x^{(y^z)}$
$x(-y)$	$x*(-y)$
$3.7+12$	$3*7+12$
$2y+5z$	$2*y+5*z$
$\frac{15+4}{27-10}$	$(15+4) / (27-10)$
$(15)^x^3$	$(15^x)^3$
$\frac{a+b}{3.x}$	$(a+b) / (3*x)$
$(x)^{3.5}$	$x^{(3*5)}$
$-5x+4y$	$(-5*x) + (4*y)$
$x.(-2y)$	$x*(-2*y)$

2. Mantık İşlemleri ve Operatörleri

<u>İşaret</u>	<u>Anlamı</u>
AND	Ve
OR	Veya
NOT	Değil
XOR	Değil veya
IMP	Dahil etme
EQV	Eşitlik

3. İşlem Öncelik Sıraları ve Parantezli İşlemler

Matematikte olduğu gibi aynı sırayı takip eder.

<u>Sembol</u>	<u>Anlamı</u>	<u>Örnek</u>
^ veya **	Üs (Kuvvet)	7^3 veya 7*7*7
*	Çarpma	5*8
/	Bölme	30/6
+	Toplama	20+8
-	Çıkarma	10-5

Yukarıdaki tabloda verilen sıra geçerlidir. Ancak çarpmanın, bölmeye; toplamının çıkarmaya önceliği yoktur. Bu türden öncelikleri ayarlamak için, önceliği olan işlem parantez içine alınır. İç içe parantezlerde, en içteki parantezin önceliği vardır. Önceliği aynı olan işlemler, soldan sağa doğru yapılır.

Aşağıdaki örneği dikkatle inceleyecek olursak parantezin yerinin önemini, çıkan sonucun nasıl değiştiğini görerek kavrayabiliriz.

2*3+4*5	→	6+20	→	26
(2*3)+(4*5)	→	6+20	→	26
(2*3+4)*5	→	(6+4)*5	→	50
2*(3+4)*5	→	2*7*5	→	70

4. Karşılaştırma Sembolleri:

<u>Sembol</u>	<u>Anlamı</u>	<u>Örnek</u>
=	Eşit	A=B
<>	Eşit Değil	A<>B
<	den küçük	A	den büyük	A>B
<= veya =<	den küçük veya eşit	A<=B veya A== veya =>	den büyük veya eşit	A>=B veya A=>B

Karşılaştırma operatörleri koşul sürmede kullanılırlar. Birkaç örnek yapalım:

```
IF A>=0 AND A<=20 THEN GOTO 130
IF A=1 OR A=5 THEN GOTO 400
IF NOT A=0 THEN GOTO 200      gibi...
```

B. PROGRAMLAMA İLE İLGİLİ GENEL BİLGİLER

Bu bölümde programlama ile ilgili bilgileri öğreneceğiz. Şunu hiç bir zaman unutmamalıyız. Bütün programların mantığı aynıdır. Yalnızca kodlama sistemi değişiktir. Bir programlama dilini bilen kişi diğer programlama dillerini öğrenmekte zorlanmaz. Her ne kadar biz burada BASIC öğreneceksek de, konuların anlatımı programlamanın mantığını kavrama açısından ele alınmıştır. Yani burada yalnız BASIC öğrenmeyeceğiz, programlamanın mantığını da kapatacağız.

1. PROGRAM, PROGRAMCI ve ANALİST

Bilgisayarda bir sonuç elde etmek amacı ile, bilgisayara verilen bilgilerin işlenmesi için belli bir mantığa dayalı olarak verilen komutlar dizisine, **program** denir.

Yüksek düzeyli dillerden biri ile, o dilin ve bilgisayarın mantığına dayalı olarak yapılan programın bilgisayara verilmesi ve çalıştırılmasına **programlama**, bu işi yapan kişiye de **programcı**, denir.

Analist: Çözümü gerekli problemler için gerekli araştırmayı yapıp, o problemle ilgili bilgileri sıraya koyan, düzenleyen kişidir.

2. PROGRAMLAMA SAFHALARI

Bir program yaparken ve bunun sonucunu alırken izlememiz gereken bir yol vardır. Burada nasıl bir yol izlememiz ve ne yapmamız gerektiğini göreceğiz:

a. Problemi Tanıma

Yapılacak problem hakkında, geniş bilgi sahibi olmamız gerekir. Program yapmak için, programcı olmak veya programlamayı bilmek yetmez; çözülecek problem konusunda iyi bir bilgi sahibi olmamız gerekir. Bir programcı ilk iş olarak problemi iyi tanımalıdır.

b. Çözüm Yollarını Tasarlama

Bir problemi iyice tanıdıktan, konu üzerinde yeterince hakim olduktan sonra, problemi nasıl, hangi yöntemle çözeceğimizi tasarlamamız gerekir. Bunun için de bir yol belirlemeliyiz. Bu yol hakkında bir karara vardıldıktan sonra, diğere aşamaya geçmeliyiz.

c. Programı Kodlama

Buraya kadar elde ettiğimiz bilgiler ışığında, çözüm için yüksek düzeyli dillerden biri ile, o dilin kurallarına uyararak kodlamalıyız.

d. Programı Test Etme

Kodlama işini bitirdikten sonra, program çalıştırılır, yazım ve mantık hataları varsa bunlar düzeltilir ve kesin bir sonuca ulaşılır.

e. Programı Belgeleme

Programdan istenilen sonuç alındıktan sonra çevre, bellek birimlerinden birine kayıt edilerek, program saklanır ve belgelenir. Ya da yazıcıdan programın listesi alınır.

3. ALGORİTMA

a. Algoritmanın Tanımı ve Kuralları

Algoritma: Bir problemin çözümü için takip edilecek yolun belirlenmesine, **algoritma** denir.

Bir problemin çözümünde değişik kişiler, değişik yöntemler izleyebilirler: Önemli olan gidilen yolun doğru ve istenilen sonuca ulaşılmasıdır. Bu da değişik kişilerin, değişik algoritma hazırlayacağı anlamına gelir.

Kısaca, günlük yaşantımızda yaptığımız, yapacağımız bütün işlerin, ister istemez kafamızda algoritmasını yaparız. Burada bir problemin çözümü için, adım adım takip edilecek yolu belirlemeğe ve hazırlanan bir çeşit taslağa gereksinmemiz olacaktır. Bu da algoritma diye tanımlanır.

Algoritmada uyulması gereken kurallar şunlardır:

1. **BAŞLA** komutu ile başlanmalıdır.
2. Verilen komutlar kısa, öz ve emir cümlecikleri şeklinde olmalıdır.
3. Her satıra, bir satır numarası verilmelidir. Satır numaraları küçükten büyüğe doğru gitmelidir.

1	10	5	
2	20	10	
3	30	12	
4	40	13	
5	50	18	gibi...

4. Kullanılan değişken adları kısa, net ve tanınabilir olmalıdır.
5. Algoritmada bazı satırların işlem görmeden atlanmasını ve işleme diğer satırdan başlamasını istiyorsak **GİT** komutu ile gideceği satır numarasına gönderilir ve işlem akışı o satırdan itibaren sürer.
6. Programın sonuna (son satırına) bittiğini belirtmek için **DUR** komutu verilmelidir.

b.Algoritma Örnekleri

Örnek 1: Döviz hesabı yapan programın algoritması.

- | | | |
|-------------------------|---|-----------------|
| 1- Başla. | | |
| 2- Döviz miktarını gir. | → | DM |
| 3- Günlük kuru gir. | → | GK |
| 4- TL değerini hesapla. | → | TL=DM*GK |
| 5- Sonucu yaz. | → | TL=? |
| 6- Dur. | | |

Örnek 2: Yıl içinde yapılan üç yazılımin ortalamasını bulan programın algoritması.

- | | | |
|------------------------|---|---------------------------|
| 1- Başla. | | |
| 2- 1. Yazılıyı gir. | → | Y1 |
| 3- 2. Yazılıyı gir. | → | Y2 |
| 4- 3. Yazılıyı gir. | → | Y3 |
| 5- Ortalamayı hesapla. | → | ORT= (Y1+Y2+Y3) /3 |
| 6- Ortalamayı yaz. | → | ORT=? |
| 7- Dur. | | |

Örnek 3: Faiz hesabının algoritması.

- 1- Başla.
- 2- Kapitali oku. → **K**
- 3- Süreyi oku. → **N**
- 4- Faiz fiyatını oku. → **T**
- 5- Faizi hesapla. → **F= (K*N*T) /1200**
- 6- Faizi yaz. → **F=?**
- 7- Dur.

Örnek 4: Yamuğun alanını hesaplayan programın algoritması.

- 1- Başla.
- 2- Kısa kenarı gir. → **C**
- 3- Uzun kenarı gir. → **A**
- 4- Yüksekliği gir. → **H**
- 5- Alanı hesapla. → **ALAN= (A+C) *H/2**
- 6- Alanı yaz. → **ALAN=?**
- 7- Dur.

Örnek 5: Ücret hesabının algoritması.

- 1- Başla.
- 2- Günlük ücreti gir. → **GU**
- 3- Çalıştığı günü gir. → **CG**
- 4- Brüt ücreti hesapla. → **BU=GU*CG**
- 5- Sigorta primini hesapla. → **SP=BU*0.14**
- 6- Özel indirimini hesapla. → **OZIND=CG*3000**
- 7- Damga vergisini hesapla. → **DV=BU*0.004**
- 8- Gelir vergisi matrahını hesapla. → **GVM=BU- (SP+GV+DV)**
- 9- Gelir vergisini hesapla. → **GV=GVM*0.25**
- 10- Net ücreti hesapla. → **NU=BU- (SP+GV+DV)**
- 11- Sonuçları yaz. → **BU , SP , GV , DV , NU**
- 12- Dur.

Örnek 6: Klavyeden M girilince MERHABA, G girilince GÜNAYDIN, başka bir harf girilince yanlış giriş yaptınız yazıp başa dönen programın algoritması.

- 1- Başla.
- 2- Bir harf gir. → **HARF**
- 3- Eğer HARF=M ise 6. satıra git.
- 4- Eğer HARF=G ise 7. satıra git.
- 5- YANLIŞ GİRİŞ YAPTINIZ yaz. Başa dön.
- 6- MERHABA yaz. Dur.
- 7- GÜNAYDIN yaz. Dur.

Örnek 7: Birden 100'e kadar sayıları toplayan programın algoritması.

- 1- Başla.
- 2- SAYI=0 değerini al.
- 3- Sayıları bir artır. → **SAYI=SAYI+1**
- 4- Sayıları topla. → **TOPLAM=TOPLAM+SAYI**
- 5- Eğer SAYI=100 ise 7. satıra git.
- 6- 3. satıra git.
- 7- Sonucu yaz. → **TOPLAM=?**
- 8- Dur.

Örnek 8: İstenilen sayıda girilen kişilerin yaş ortalamasını bulan programın algoritması.

- 1- Başla.
- 2- Sayaç başlat. → **S=0**
- 3- Yaş gir. → **YAS=?**
- 4- Eğer YAS=0 ise 8. satıra git.
- 5- Sayacı 1 artır. → **S=S+1**
- 6- Yaşları topla. → **YTOP=YTOP+YAS**
- 7- 3. satıra git.
- 8- Ortalamayı hesapla. → **ORT=YTOP/S**
- 9- Yaş ortalamasını yaz. → **ORT=?**
- 10- Dur.

Örnek 9: Bilgi yarışmasına katılan 5 öğrenciden, birinci olan öğrencinin puanını ve adını yazan programın algoritması.

- 1- Başla.
- 2- Sayacı sıfırla. → **S=0**
- 3- En büyük değişkeni sıfırla. → **ENB=0**
- 4- Adı Soyadı gir. → **ADŞ=?**
- 5- Puanı gir. → **PUAN=?**
- 6- Sayaç artır. → **S=S+1**
- 7- Eğer puan en büyükten büyükse, en büyüğü puana eşitle. Koşul tutuyorsa, Adını yeni bir değişkene ata. → **AADŞ=ADŞ**
- 8- Girilen öğrenci sayısı 5 ise, 10. satıra git.
- 9- Yeni giriş için 4. satıra git.
- 10- Birinci gelen öğrencinin adını yaz. → **AADŞ=?**
- 11- Puanını yaz. → **ENB=?**
- 12- Dur.

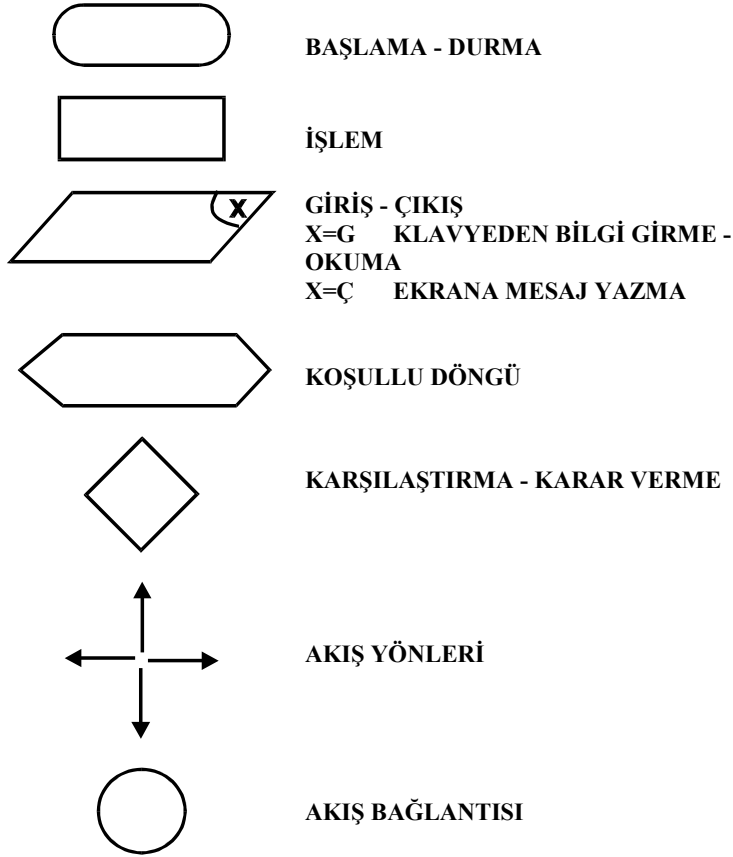
4. AKIŞ ŞEMALARI (DİYAGRAMLARI)

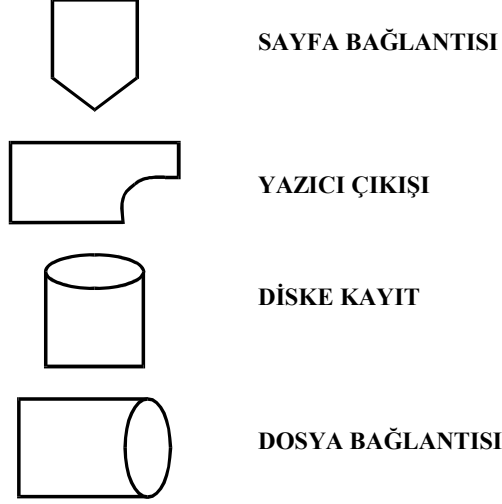
a. Akış Şemasının Tanımı

Problem çözümünde izlenecek yolun belirlenmesi için yapılacak algoritmanın şekillerle gösterilmesine **akış şeması** denir.

Akış şemalarını, çizmek için aşağıdaki şekil ve semboller kullanılır:

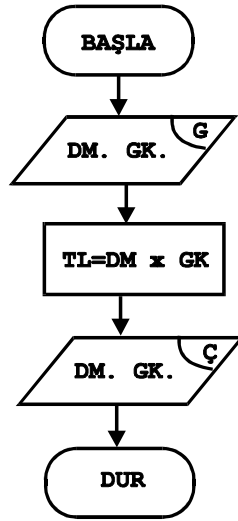
b. Akış Şeması Sembolleri



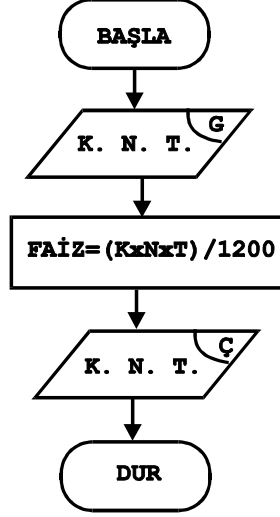
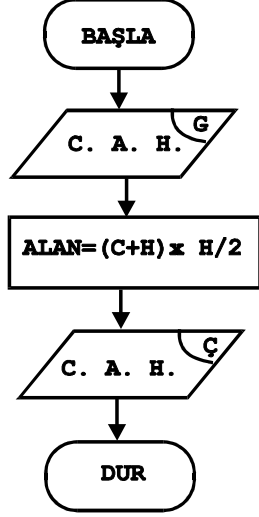


c. Akış Şeması Örnekleri

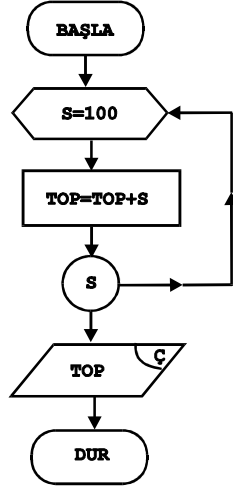
Örnek 1: Klavyeden DÖVİZ MİKTARI ve GÜNLÜK KUR girildiğinde cebimizdeki döviz TL'ye çeviren programın akış şeması.



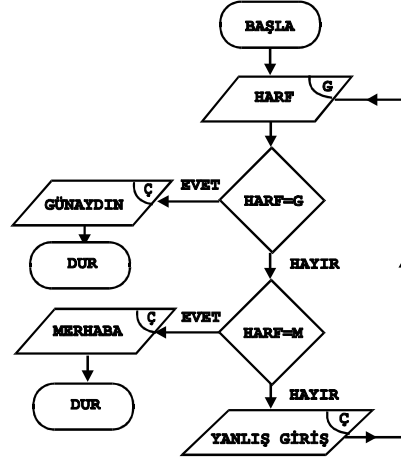
Örnek 2: Yamuğun alanını hesaplayan programın akış şeması. **Örnek 3:** Faiz hesabının akış şeması.



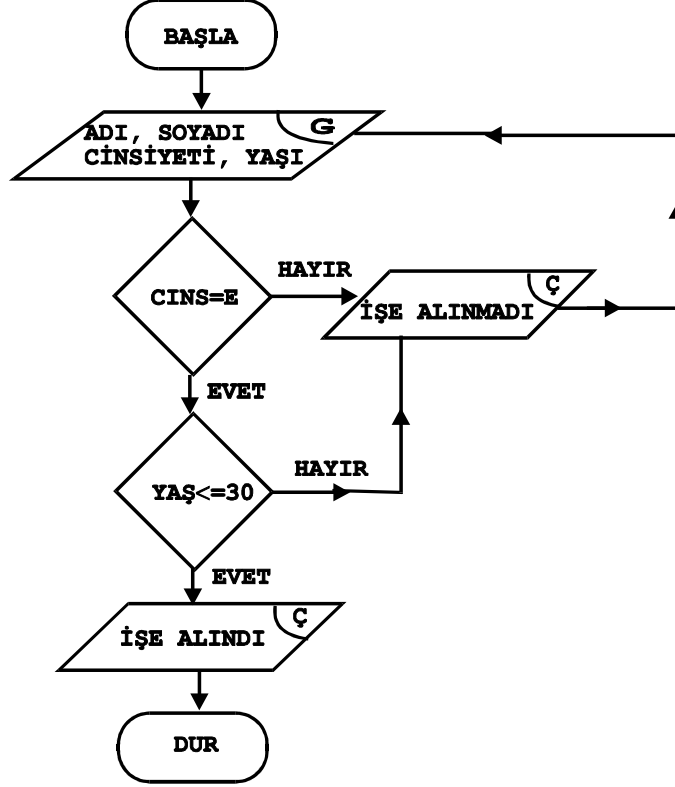
Örnek 4: 1'den 100'e kadar sayıları toplayan programın akış şeması.



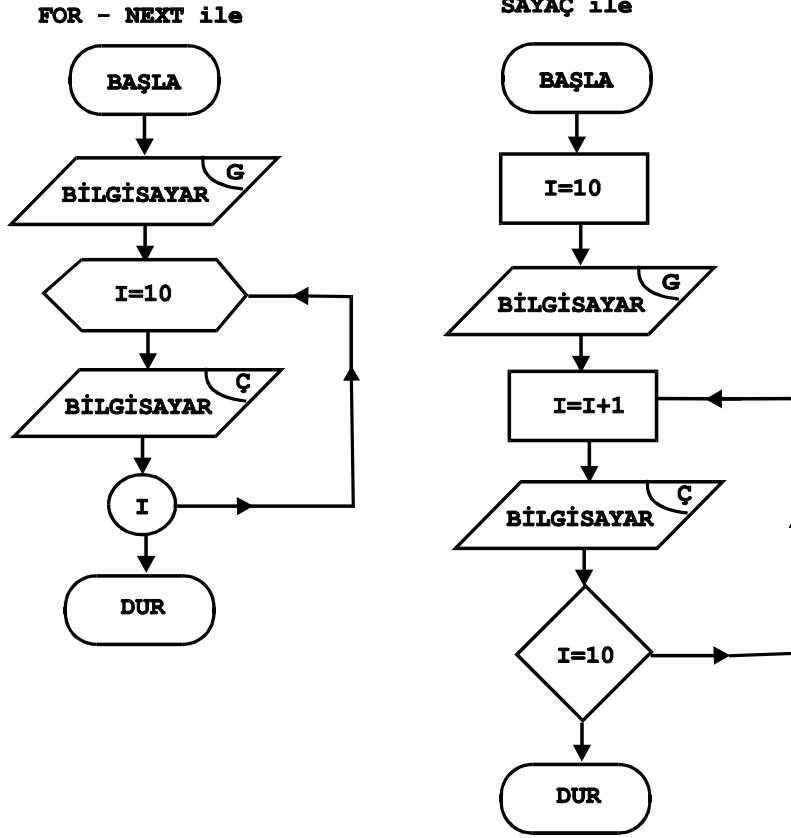
Örnek 5: Klavyeden M girilince MERHABA, G girilince GÜNAYDIN yazan programın akış şeması.



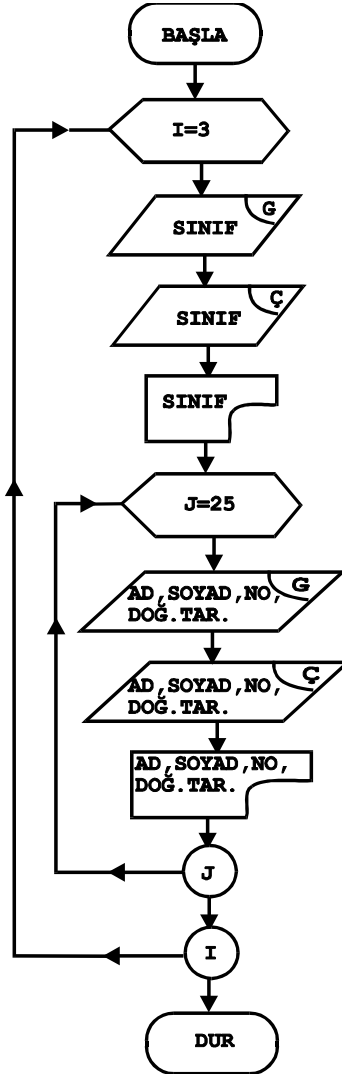
Örnek 6: Bir işyerinde işe alınacaklarda aranan koşul erkek olmak ve 30 yaşından büyük olmamaktır. Buna göre yapılacak programın akış şeması.



Örnek 7: Ekranı 10 kez BİLGİSAYAR yazdıran programın akış şeması.



Örnek 8: Bir okulun 3 sınıfında 25'er öğrenci bulunmaktadır. Bu öğrencilerin ADI SOYADI, NO, DOĞUM TARİHİ'ni ekran ve yazıcıya aktaran programın akış şeması.



5. BİLGİSAYARA DEĞERLERİN TANITIMI

a. Sabitler

Programın çalışması sırasında değerleri değişmeyen, değer ve ifadelerdir. İkiye ayrılırlar:

1) Sayısal Sabitler: Yalnızca sayıları ifade eder.

i. Tam Sayı Sabitler: -32768 ile +32767 sayıları arasındaki sayılardır.

ii. Sabit Noktalı Sayılar (Reel Sayılar): İçinde ondalık nokta bulunan sayılardır.

0.75 25.5 -11.675 gibi.

iii. Kayan Noktalı Sayılar (Üslü Sayılar): Exponansiyonel Sayılar da denir. **E** harfi ile tanımlanırlar. Pozitif ve negatif üslü sayıları kapsar.

Bu tip sayılarda **E** harfinden önceki sayılar **Mantis** kısmını, **E** harfinden sonraki sayılar da **Üs**'sü gösterir. E'nin yanında bulunan sayı noktanın ne kadar sağa veya sola kayacağını belirtir. Bu sayının önündeki işaret (+) ise nokta sağa, (-) ise nokta sola, o sayı kadar kayacak demektir.

Örnek:

52.45E+5'in matematiksel karşılığı, nokta sağa 5 kayacağından **5245000**'dir
52.45E-5'in matematiksel karşılığı, nokta sola 5 kayacağından **0.0005245**'dir.

25 . 05E+3=25050 3 . 5E+4=35000
25 . 05E-3=0 . 02505 3 . 5E-4=0 . 00035

2) Alfasayısal Sabitler: **Katar** veya **String** de denir. Sayısal ve alfabetik karakterlerden oluşur. "... " (Çift Tırnak) içinde yazılırlar.

"HÜSEYİN UYSAL" "5 HAZİRAN 1986" gibi.

b. Değişkenler

Programın çalışması sırasında, değişebilen değer ve ifadelerin adlandırılmasıdır. Ya da bu bilgilerin depolanacağı adres de diyebiliriz.

Bir değer ya da ifadeyi program içerisinde depolamak için, değer ve ifadenin depolanacağı değişkenlere ad verilir. Bu verilen adlar doğrultusunda değerler depolanır.

Değişken Adı Verirken Dikkat Edilecek Noktalar:

- ✓ Değişken adı olarak BASIC deyimleri kullanılamaz. **GOTO, ON, OR, IF** gibi.
- ✓ Değişken adı verirken özel karakterler kullanılmaz. **#, @, &, ?** gibi.
- ✓ Değişken adı verirken Türkçe karakterler (**ş,ç,ö,ü,ı** gibi...) kullanmamakta yarar vardır. Bu karakterler bilgisayar için özel karakterlerdir.
- ✓ Değişken adında boşluk kullanılmaz. **AD SOYS** yerine **ADSOYS** yazılmalıdır.
- ✓ Değişken adı verirken, depolanmak istenen değer ya da ifadenin kısaltmasının verilmesi karışıklığı önlemek ve kolay anımsamak açısından uygundur.
- ✓ Değişken adı verirken rakamla başlamamalıdır. **1ADS** yerine **AD1S** yazılmalıdır.

1) Sayısal Değişkenler: Sayısal değerlerin depolanacağı ya da atanacağı değişebilen sayısal değerlerin adlandırılmalarıdır. Sayısal değişkenlerin sonuna sayının durumuna göre **%, !, #** işaretlerinden biri konur. Bu işaretler konmadığı takdirde, tek duyarlıklı sayısal değişken olarak kabul görür.

- %** Tam sayı değişkenler için. **SAYI%, FİYAT%, TUTAR%** gibi.
- !** Tek duyarlıklı sayı değişkenler için. **SAYI!, FİYAT!, TUTAR!** gibi.
- #** Çift duyarlıklı sayı değişkenler için. **SAYI#, FİYAT#, TUTAR#** gibi.

2) Alfasayısal Değişkenler: Alfabetik ve sayısal karakterlerden oluşan, değişen ifadeleri adreslemek için yapılan adlandırmadır. Bu tür değişkenlerin sonuna, sayısal olmadığını belirtmek ve bilgisayarın, o değişkenin alfasayısal bir değişken olduğunu anlaması için **\$** işareti konur. Bu değişkenlere depolanan ifadeler **"..."** içinde yazılırlar.

DTARIH\$="9 MART 1961", AD\$="GÜVEN TANIŞ" gibi.

C. PROGRAMLAMA DİLLERİ

Bilgisayarlar, ikili sayılı düzenine göre çalışırlar. Bilgisayarların merkezi işlem biriminin kullanabileceği biçimde, ikili sayı sistemindeki komut kodlarının amaca göre sıralanması gerekir. Buna makine dili adı verilir. Kullanılması, yazılması ve hata olduğunda bu hatanın bulunması ve düzeltilmesi çok güçtür. Bilgisayarın problemi çözebilmesi için onun anlayacağı bir dille program yapmak gerekir.

Yukarıda anlattığımız, makine dilinin bu güçlüğü nedeniyle daha kolay, kolaylıkla anlaşılabilen diller doğmuştur.

Bu diller **Alçak Düzeyli** ve **Yüksek Düzeyli** olmak üzere ikiye ayrılır.

1. Alçak Düzeyli Diller

Bu tür dillerde komutlar ikilik sayı sistemi ile verilir. Bildiğimiz gibi ikili sayı sayı sistemi **0** ve **1** rakamlarından oluşuyordu. Bu nedenle yalnızca iki simge ile program yapmak zordur ve yanlış yapıldığında, yanlış bulması da aynı derecede güçtür. Bu dile örnek olarak **ASSEMBLY** dilini verebiliriz.

2. Yüksek Düzeyli Diller

Bu diller, makinenin yapısından tamamen bağımsızdırlar. Konuşma diline yakın olmalarından dolayı program yapmak kolaydır. Bu dillerle yapılan programlarda hata oranı az ve hatayı bulmak kolay olduğundan programcılar arasında daha çok ilgi görürler ve kullanılırlar.

Başlıca Yüksek Düzeyli Diller:

FORTRAN: **F**ormula **T**ranslate sözcüklerinden oluşmuştur. Formül Çevirici anlamındadır. Daha çok matematiksel hesaplamalarda ve mühendislik alanlarında kullanılır. İlk resmi versiyonu **1957** yılında gerçekleştirilmiştir. **1977** yılında **ANSI** tarafından son değişiklikleri yapılarak standart ve esnekliğe kavuşmuştur. (**ANSI**=Amerikan Standartları Enstitüsü)

COBOL: **C**ommon **B**usiness-**O**riented **L**anguage sözcüklerinden oluşmuştur. Bu dil daha çok ticari amaçlı programlar için kullanılır. Dünya çapında kabul edilmiş bir dildir. Bu dil yaratılırken amaçlanan, komutların İngilizce'ye yakın olmasıdır. Kaynak program yazıldıktan sonra **COBOL** derleyicisi tarafından derlenir.

PL/1: Programming Language-**1** sözcüklerinden oluşmuştur. Matematiksel ve bilimsel amaçlı programlar için kullanılır.

PASCAL: Fransız matematikçi **Blaise Pascal**'ın adını taşır. İlk versiyonu **1968** yılında ortaya çıkmış, İsviçre'li **Niklaus Wirth** tarafından geliştirilmiştir. Matematiksel çalışmalar için gerçekleştirilmiştir. **PL/1** ve **FORTTRAN**'ın özelliklerini taşır. Yapısal bir dildir. Program düzenlemesi ve formatında belli kuralları vardır. Çok gelişmiş bir dildir.

RPG: Report Program Generator sözcüklerinden oluşmuştur. Rapor vb. dökümanlar ve ticari tabloların hazırlanmasında kullanılır. Matematiksel ve grafik çizimi için elverişli değildir.

LOGO: Grafik çizimi için kullanılır. Daha çok eğitim amaçlı programların hazırlanması için gerçekleştirilmiştir.

BASIC: Beginner's All-purpose Symbolic Intruction Code sözcüklerinin ilk harflerinden oluşmuştur. Programlamaya yeni başlayanlar için geliştirilmiştir. Kolay öğrenilen ve yaygın kullanımı olan bir dildir.

C: En son geliştirilen dillerden biridir. Dosya işlemleri ve matematiksel işlemlerle ilgili fonksiyonları içerir. Yapısal ve genel amaçlı bir dildir. **1972** yılında **Dennis Ritchie** tarafından **XENIX** işletim sistemi için geliştirilmiştir. **UNIX** işletim sistemi ve **C derleyicisi**, **C** diliyle yazılmıştır.

ADA: Sayısal ve sistem programcılığında kullanılan bir programlama dilidir. Genel amaçlı bir dildir.

Yukarıdaki dillerin dışında **APL (1968)**, **ALGOL (1959)**, **PROLOG (1972)** gibi... yüksek düzeyli diller bulunmaktadır.

SORULAR

1. Program nedir?
2. Programlama, Programcı ve Analist'in tanımını yapınız.
3. Algoritma nedir? Uyulması gereken kurallar nelerdir?
4. Programlama safhalarını sayınız.
5. Akış şemalarında Başla, Karşılaştırma ve Ekran yazma sembolleri nelerdir?
6. Bir malın satış ve maliyet fiyatı verildiğinde, karı hesaplayan programın algoritmasını ve akış şemasını yapınız.
7. Verilen dört sayının ortalamasını alan programın algoritmasını ve akış şemasını yapınız.
8. Bir işyerinde bulunan işçilerin ücretlerini hesaplayan programın algoritmasını ve akış şemasını yapınız.
9. Yamuğun alanını hesaplayan programın algoritmasını ve akış şemasını yapınız.
10. Sabit ve Değişken nedir? Aralarındaki farkları açıklayınız.
11. Değişken adı verirken uyulması gereken kurallar nelerdir?

BÖLÜM IX.

BASIC PROGRAMLAMA DİLİ

A. BASIC DİLİNE GİRİŞ

1. Basic Dilinde Program Yazımı İçin Gerekli Ön Bilgiler:

a. Karakterler: Basic programlama dilinde kullanılan başlıca karakterler şunlardır:

1) **Harfler:** (A....Z) büyük harfler , (a....z)'ye kadar bütün küçük harfler.

2) **Sayılar:** 0'dan 9'a kadar bütün sayılar.

3) **Özel Karakterler:** Yazım dilinde kullanılan karakterlerin dışındaki karakterlerdir. Bunların hepsinin basic dili için özel bir anlamı vardır.

	→	Boşluk
=	→	Eşit İşareti
+	→	Artı İşareti
-	→	Eksi İşareti
*	→	Çarpı İşareti
/	→	Bölü İşareti
(→	Açma Ayracı (Parantez)
)	→	Kapama Ayracı
!	→	Ünlem İşareti
\$	→	Dolar İşareti
#	→	Sayı (Number) İşareti
^	→	Üst İşareti
%	→	Yüzde İşareti
.	→	Nokta İşareti
,	→	Virgül İşareti
;	→	Noktalı Virgül İşareti
:	→	Üst Üste İki Nokta İşareti
"	→	Çift Tırnak İşareti
'	→	Kesme (Tek Tırnak) İşareti
@	→	Yer İşareti
&	→	Birleşme (Ampersand) İşareti
?	→	Soru İşareti
\	→	Ters Bölme (Slaş)
<	→	Küçüktür İşareti
>	→	Büyüktür İşareti
ENTER	→	Terminalden giriş

2. ÖZEL KARAKTERLERİN AÇIKLAMALARI

Yukarıda verdiğimiz işaretlerin bazılarının, **BASIC**'te günlük yaşamda kullandığımız işlevlerinden farklı işlevleri vardır. Bu işaretler ve kullanılışları şöyledir:

. (Nokta) : Cümle bitimlerinde kullanıldığı gibi, kesirli sayıları ayırmada da kullanılır.

<u>Günlük Yaşamda</u>	<u>BASIC'te</u>
5,25	5.25
0,15	0.15
75,5	75.5

; (Noktalı Virgül) : Çıktıların yanyana yazılması için kullanılır.

<u>Yazılışı:</u>	<u>Çıktısı:</u>
10 PRINT "BIR"; "IKI"; "ÜÇ";	BIRIKIÜÇ
20 PRINT 10;45;-18	10 45-18

, (Virgül) : Noktalama işareti olarak ve çıktıları aralıklı yazmak için kullanılır. Ekranı **16 karakterlik** bölümlere ayırarak yazdırır.

Yazılışı:
10 PRINT "BIR", "IKI", "ÜÇ"
20 PRINT 10,45,18

Çıktısı:

BIR 10	IKI 45	ÜÇ 18
16 Karakter	16 Karakter	16 Karakter

" (Çift Tırnak) : Alfasayısal bilgilerin yazılması için kullanılır.

Örnek:
"ALI BAKI USTA" "GÖRELE, GİRESUN" "23 HAZİRAN 1990" gibi.

\$ (Dolar İşareti) : Alfasayısal değişkenlerin tanımlanmasında kullanılır.

Örnek:
AD\$="SELÇUK AKATA" TARIH\$="19 MAYIS 1919"

: (Üst Üste İki Nokta): Bir program satırına, birden çok komut ve deyim yazmak için kullanılır.

```
10 CLS:KEY OFF:CLEAR
20 LOCATE 3,20:PRINT "ENVER KOYUNCU"
```

? (Soru İşareti) : **PRINT** komutu yerine kullanılır. **INPUT** komutunun bulunduğu satırda ? işareti, giriş belirtmek amacıyla karşımıza çıkar.

```
10 CLS
20 KEY OFF
30 CLEAR
40 LOCATE 3,20
50 PRINT "ENVER KOYUNCU"

10 CLS:KEY OFF:CLEAR
20 LOCATE 3,20:PRINT "ENVER
KOYUNCU"
yerine
```

Bu programı çalıştırdığımızda, biz ? işareti koymadığımız halde karşımıza, kendiliğinden çıkacaktır. **10.** satırda **INPUT** komutunu kullandığımız için bizden bilgi girmemiz istenmektedir. **20.** satırda ise **PRINT** yerine soru işaretini kullanarak kısa yoldan yazmak istedik. Tekrar programın listesini alacak olursak **20.** satırdaki ? işaretinin yerine **PRINT** komutunun geldiğini görürüz.

```
RUN
ADINIZI GIRINIZ? NALAN TANIŞ
NALAN TANIŞ
Ok
```

' (Tek Tırnak İşareti) : **REM** komutu yerine kullanılır.

```
10 REM BU PROGRAM 30 TEMMUZ 1990 TARİHİNDE HAZIRLANMIŞTIR.
yerine,
10 ' BU PROGRAM 30 TEMMUZ 1990 TARİHİNDE HAZIRLANMIŞTIR.
yazılabilir.
```

= (Eşittir İşareti) : Değişkenlere değer atamada kullanılır.

```
A=25 CINS$="KITAP" gibi.
```

% (Yüzde İşareti) : Bilgisayarda tam sayılar için kullanılan değişkenleri tanımlamada ve **PRINT USING** komutuyla kullanılır. Bilgisayar dilinde **Tam Sayılar**'ın değerleri **-32768** ile **+32767** arasındaki sayılardır.

```
A%=4587 S%=24500 gibi.
```


! (Ünlem İşareti) : Tek duyarlıklı sayılar için kullanılan değişkenleri tanımlamada ve **PRINT USING** ile birlikte kullanılır. **Tek Duyarlıklı Sayılar 7 hane'yi** geçmeyen reel sayıdır.

A!=35888 S!=456982 gibi.

Program içerisinde tek duyarlıklı sayılarda bu tanımlamayı yapmazsak ve sayı **+32767**'yi geçiyorsa, bilgisayar otomatik olarak ünlem işaretini koyar. **A=35888** şeklinde yazdığımızda bu **A=35888!** şekline dönüşür.

(Sayı İşareti) : Çift duyarlıklı sayıları içeren değişkenleri tanımlarken ve **PRINT USING** deyiimiyle birlikte kullanılır. **Çift Duyarlıklı Sayılar 7 hane'den çok 17 haneden** az olan sayılardır.

İşleyişi ! işareti gibidir. # işareti konulmazsa bilgisayar kendiliğinden bu işareti koyar.

A#=267482353 S#=8743754573467 gibi.

B. BİLGİSAYARI AÇMA ve BASIC ORTAMINA GEÇİŞ

Bilgisayarımıza **MS-DOS** işletim sistemini takar ve bilgisayarı açarız. Bundan sonra, kullanacağımız **BASIC** yorumlayıcısı çalışma disketimizde yoksa, disketimize ilk olarak **BASIC** yorumlayıcısını yüklemeliyiz. **BASIC** yorumlayıcısını yükledikten sonra, artık **BASIC** programlama yapabiliriz. İşletim sisteminden **BASIC** ortamına geçmek için, çalışmak istediğimiz **BASIC** yorumlayıcısının adını, gerekiyorsa anahtarlarını girerek **BASIC** ortamına geçebiliriz. Piyasada değişik adlarla çeşitli **BASIC** yorumlayıcı ve derleyicileri bulunmaktadır. **TURBO BASIC, QUICK BASIC, QBASIC** gibi yorumlayıcı ve derleyiciler, göreceğimiz Basic'le biraz farklıdır, mantık olarak aynıdır. Bizim konumuzu ilgilendiren Basic Yorumlayıcıları şunlardır: **BASIC, BASICA, GWBASIC**.

Yazılışı:

A: \>BASIC ↵

Ya da

**A: \>BASIC [Dosya Adı] [/F:Dosya Sayısı]
[/S:Tampon Bellek Büyüklüğü]**

Yazıp ekranın değişmesini bekleriz. **BASIC** ortamına geçince ekran değişerek aşağıdaki şekli alır:

```
GW-BASIC 3.20
(C) Copyright Microsoft 1983,1984,1985,1986
60332 Bytes free
Ok

1LIST 2RUN← 3LOAD" 4SAVE" 5CONT← 6,"LPT1 7TRON← 8TROFF← 9REY 0SCREEN
```

Aşağıdaki anahtarları da gerektiği durumlarda kullanmamız gerekir.

Basic Anahtarları:

/F:Dosya Sayısı: Kullanılacak basic programının çalıştırılması sırasında açılacak en çok dosya sayısını belirler. Bu anahtar belirtilmezse, kullanılacak dosya sayısı **3** kabul edilir.

/S:Tampon Bellek Büyüklüğü: Dosya organizasyonunda kullanılacak tampon bellek büyüklüğünü belirler. Data dosyalarında **OPEN** deyimiyile belirtilen dosya uzunluğu bu değeri geçmemelidir. Bu anahtar kullanılmazsa değer **128** kabul edilir.

GWBASIC kullanıyorsak geçmek şu şekilde yazarız:

```
A>GWBASIC ↵
```

C. BASIC'DE KLAVYE KULLANIMI

İşletim sistemi konusunda klavyenin kullanımını görmüştük. Ama bu **MS-DOS** için geçerli idi. Klavyede bazı tuşların programın özelliğine göre tanımlanabileceğini, bu tür değişikliklerin kullanımı kolaylaştırdığını da söylemiştik. Bu söylediklerimiz çoğu editörler için de geçerlidir. Bu nedenle **BASIC** programlama içinde çalışırken, **BASIC** yorumlayıcısını (Interpreter) yüklediğimizde de program yazmak için, klavye kullanım kolaylıkları sağlanmıştır. Şimdi bunları göreceğiz:

Bu komutları görmek için aşağıdaki basic programını, basic ortamına geçerek yazınız. Yazarken bir alt satıra geçmek ve satırı belleğe göndermek için her satırın sonunda **Enter** tuşuna basınız.

```
10 CLS
20 FOR S=1 TO 20
30 TOPLAM=TOPLAM+S
40 PRINT S, TOPLAM
50 NEXT S
60 STOP
70 PRINT
80 PRINT
90 PRINT
100 PRINT
110 PRINT "PROGRAMI HAZIRLAYAN: GÜVEN TANIŞ"
120 PRINT "BU PROGRAM KLAVYE KULLANIMINA ÖRNEKTİR."
130 FOR A=1 TO 5
140 PRINT A
150 NEXT A
160 PRINT "TEŞEKKÜR EDERİM"
170 END
```

Bu programı yazdıktan sonra klavye kullanımına geçebiliriz.

1. FONKSİYON TUŞLARI

BASIC programlamada, çalışma kolaylığı sağlayan bu tuşların bazıları yalnızca tuşa basmakla, bazıları da tuşun ardından **ENTER**'e basılmasıyla işler. Ekranın altında fonksiyon tuşlarının hangi amaçla kullanılacağını görebilirsiniz. Eğer komuttan sonra ← işareti varsa direk çalışan tuştur.



LIST

Program listesini almak için kullanılır. Liste akışını durdurmak gerektiğinde **Ctrl+Num Lock** tuşlarına ya da **Pause** tuşuna basmamız gerekir. Tekrar devam etmesi için herhangi bir tuşa basmamız yeterlidir. Eğer liste akışını kesmek istiyorsak, bunu da **Ctrl+Break** tuşlarına birlikte basarak yapabiliriz.

Şimdi **F1** tuşuna basarak veya klavyeden **LIST** yazalım ve **Enter** tuşuna basalım. Az önce yazmış olduğumuz programın listesinin ekrana yeniden geldiğini göreceğiz.

LIST komutunun çeşitli kullanım biçimlerini görelim. Aşağıdaki liste alma örneklerini tek tek deneyiniz ve sonucunu görünüz.

LIST	↵	Programın tamamını listeler.
LIST 100	↵	Yalnızca 100. satırı gösterir.
LIST -80	↵	Programın başından 80. satıra kadar listeler.
LIST 80-120	↵	80. ve 100. satırlar da dahil arası listeler.
LIST 120-	↵	120. satırdan sonuna kadar listeler.

Programın listesini **Yazıcı**'dan almak istersek **LIST** komutunun başına **L** harfini yazarak, **LLIST** şeklinde yukarıdaki kurallar doğrultusunda listeyi yazıcıdan alabiliriz.

Ya da **LIST** ile birlikte **LPT**'yi kullanarak yazıcıdan listeyi yine alabiliriz.

LIST 80-120 "LPT:" gibi.

NOT: ↵ işaretleri **ENTER** tuşuna basılacağını gösterir.



RUN

RUN komutu tek başına yazıldığında bellekte yüklü programı çalıştırır.

Yazılışı:

```
RUN [Satır No]
RUN <"Dosya Adı">[,R]
```

Bir programı çalıştırmak için, o programın bellekte olması gerekir. Yeni yazdığımız bir programı çalıştıracaksak, programımız bellekte olacağından direk çalışır. Daha önce yazmış ve diskete kaydetmiş isek, bu programı belleğe yüklememiz gerekir.

Programımız bellekte olduğu için, **F2** tuşuna basalım. Programın çalıştığını ve **1**'den **20**'ye kadar sayıların toplamını vererek, **Break in 60** yazarak bittiğini göreceğiz. **F2** tuşuna bastığımızda **Enter**'e basmadan program çalıştı. Ekranın altında **F2**'nin yanında **RUN←** işaretini görürüz. Bu **Enter**'a basmadan bu komutun çalışacağını gösterir. **F2** yerine klavyeden **RUN** yazıp **Enter** tuşuna basarak da programı çalıştırabiliriz.

Komuttan sonra, satır no verilirse, program belirtilen satır no'dan itibaren çalışmaya başlar. Bu kez aşağıdaki şekilde yazarak çalıştıralım:

```
RUN 110 ↵
```

Programın **110**. satırdan sonra çalıştığını göreceğiz. Programda belirtildiği gibi **1**'den **5**'e kadar sayıları yazacak ve **Ok** yazıp duracaktır.

Dosya adından sonra, **R** modunu yazarsak o anda **OPEN** ile açılmış bütün dosyaları kapatır.



SAVE"

Program yazıldıktan sonra diskete kayıt edilmek istendiğinde, **SAVE** komutuyla kayıt edilir. Burada bu komutun görevi **F4**'e yüklenmiştir. Yazdığımız programı kayıt etmek istiyorsak, 8 karakteri geçmeyen bir ad vererek kayıt edebiliriz.

SAVE" komutunu çıkarırız. Yanına program adını yazar ve " tırnağı kapatıp **Enter**'e basarız. Biz programımızın adını **DENEME** koyalım.

SAVE"DENEME" ↵ yaptığımızda programımız diskete kayıt edilecek ve kendiliğinden .BAS uzantısını alarak, bir BASIC programı olduğunu belli edecektir.

Aynı programı bir başka adla yeniden kaydedelim. Bu kez **F4**'e basmadan klavyeden **SAVE"ORNEK"** yazıp Enter tuşuna basalım. Bu şekilde içeriği aynı olan iki değişik adda, iki adet programımız oldu.



LOAD"

Diskette kayıtlı bulunan bir **BASIC** program çalışma dosyasını disketten çağırır.

Programın uzantısı **.BAS** ise uzantıyı yazmaya gerek yoktur. Değilse **BASIC** programı olduğundan eminsek, program adını uzantısı ile birlikte yazmalıyız.

Bu komutun kullanımını görmek için belleği temizleyelim. İmleci sol tarafa alarak **NEW** yazıp Enter tuşuna basalım. Karşımıza **Ok** mesajı çıkar. Bu işlemden sonra bellekte hiçbir şey kalmamıştır. Denemek için **LIST** ya da **RUN** komutunu kullanınız. Hiçbir şey çıkmadığını göreceksiniz.

Az önce yaptığımız programı diskete kaydetmiştik. Bu programı çağıralım:

F3 tuşuna basarak **LOAD"** mesajını alalım. Yanına **ORNEK** yazıp " işareti ile kapatarak **Enter**'e basalım. Yani **LOAD"ORNEK"** yazıp **Enter**'e basacağız. **Ok** mesajını aldıktan sonra, programın belleğe yüklendiğini anlarız. Artık programımız üzerinde istediğimiz çalışmayı yapabiliriz.



CONT

Ctrl+Break veya **STOP** komutuyla durdurulmuş programı, kaldığı yerden itibaren çalıştırır.

Programımız bellekte ise **RUN** yapıp çalıştıralım. **1**'den **20**'ye kadar sayıları topladıktan sonra **Break in 60** mesajını yazıp duracaktır. Şimdi **F5** tuşuna basarak ya da klavyeden **CONT** deyimini girerek **Enter** tuşuna basalım. Programın kaldığı yerden çalışmaya devam ettiğini ve **1**'den **5**'e kadar sayıları yazdığını göreceğiz. Aynı işlemi programı çalıştırdıktan sonra **Ctrl+Break** ile durdurarak, sonra kaldığı yerden tekrar **CONT** deyimiyile sürdürmeyi deneyiniz.



, "LPT1:"

Bu komut başka bir komutla birlikte kullanılır. 1 nolu yazıcıya aktarımı sağlar. **LIST**, "LPT1:" programın listesini yazıcıdan verir.



TRON

Programın çalışmasını satır satır takip ederek, çalışan satırın numarasını ekranda gösterir.

Programımız bellekte iken **F7** tuşuna basarak veya klavyeden **TRON** komutunu girdikten sonra çalıştırırız. Programın çalışması sırasında çalışan satırın da ekranda çıktığını göreceksiniz.

```
[20] [30]
[40]
[50] [60] [70] [120]
[40] [50]                gibi...
```



TROFF

Bilgisayarı **TRON** modundan çıkararak normal konumuna getirir. **TRON** komutunu kullandıktan sonra programlarımızı her çalıştırdığımızda, çalışan satırın numarasının ekrana yansıdığını göreceğiz. **TRON** modundan çıkmak, normal çalışma ortamına dönmek için **F8** tuşuna basın ya da klavyeden **TROFF** deyimini girin. Bundan sonra programımız çalıştırıldığında, satır numaraları ekranda görünmeyecektir.



KEY

KEY (Tuş), anahtar anlamındadır. Fonksiyon tuşlarından herhangi birine bir karakter dizisini yüklemek için kullanılır.

KEY 6,"FILES"

6 numaralı fonksiyon tuşuna **FILES** komutu yüklenir. Diskette yüklü bulunan dosyaların adlarını verir.

KEY 5, "SYSTEM"+CHR\$(13)

5 numaralı tuşa basınca **SYSTEM** komutu **ENTER**'e basmaksızın çalışır. **BASIC**'ten **DOS**'a döner.

KEY LIST Fonksiyon tuşlarının listesini ve görevlerini ekrana verir.

```
KEY LIST  ↵  
F1 LIST  
F2 RUN←  
F3 LOAD"  
F4 SAVE"  
F5 CONT  
F6 "LPT1 :"  
F7 TRON  
F8 TROFF  
F9 KEY  
F10 SCREEN 0,0,0
```

KEY OFF En alt satırda bulunan fonksiyon tuşlarının görevlerinin ekranda gösterimi kalkar. O satır da kullanıma açılır.

KEY ON Fonksiyon tuşlarının en alt satırda görüntülenmesi yeniden sağlanmış olur.



SCREEN 0,0,0

SCREEN (Ekran) komutu, grafik modundan yazı moduna geçiş için.

2. EDİT TUŞLARI



HOME (Yuva) tuşuna bastığımızda imleç, sol üst köşeye gider. **Ctrl** tuşu ile birlikte basacak olursak, ekran silinir ve imleç sol üst köşeye gider.



End: İmleci, o anda ekranda bulunan yazıların son karakterine götürür.



DEL (Silme) tuşuna bastığımızda, imlecin o an üzerinde bulunduğu karakter silinecek ve sağında bulunan bütün karakterleri kendine doğru çekecektir.

BİLGİLİKSAYAR Yazdığımızı varsayalım. Doğrusu **BİLGİSAYAR** olacaktı. Buradaki düzeltmeyi yapalım. İmleci ok tuşlarıyla **L** harfinin üzerine getirelim aşağıdaki durumu alsın.

BİLGİLİKSAYAR Şeklinde iken **Del** tuşuna **3** kez basalım ve istediğimiz düzeltmeyi yapalım. İmlecin üzerinde bulunduğu harfi silerken sağındaki karakterleri kendine doğru çektiğini görürüz.



Insert: Araya sokma modu için bu tuşa bastığımızda, imlecin çizgi imleç ise yarım büyüdüğünü, bütün imleç ise yarım küçüldüğünü görürüz. İmlecin üzerinde bulunduğu yerden sonraki karakterleri sağa kaydırırken, yazdığımız karakterleri araya sıkıştırır. İkinci basışta imleç eski konumuna gelir.

BİLGİYAR yazdığımızı varsayalım. Burada **I**den sonra **SA** harfleri eksik. İmleci ok tuşlarıyla **Y** harfinin üzerine getirelim. **Ins** tuşuna basalım İmlecin yarım büyüdüğünü ve araya sokma moduna geçildiğini göreceğiz.

BİLGİYAR Şekildeki gibi, olduktan sonra **Ins** tuşuna ve arkasından **SA** harflerine basalım. Araya **SA** harflerinin girdiğini, sağdaki harflerin sağa doğru kaydığını göreceğiz. Şimdi **Ins** tuşuna yeniden basalım ve imleci eski durumuna getirelim.



Alt tuşu ile birlikte sayısal klavye bölümünden, karakterlerin karşılığı girilirse, klavyede bulunmayan, bilgisayarda bulunan karakterlerin yazıldığını görürüz.

Alt+154	Ü harfini
Alt+148	ö işaretini
Alt+141	ı harfini
Alt+64	@ işaretini verir.

Alternatif tuşu, başka tuşlarla birlikte kullanılır. Bu tuşla birlikte alfabetik tuşlara basarsak bazı **BASIC** komutlarının yazılışını hazır olarak bulabiliriz.

TUŞ	KOMUT	TUŞ	KOMUT	TUŞ	KOMUT
A	AUTO	J	Boş	S	SCREEN
B	BSAVE	K	KEY	T	THEN
C	COLOR	L	LOCATE	U	USING
D	DELETE	M	MOTOR	V	VAL
E	ELSE	N	NEXT	W	WIDTH
F	FOR	O	OPEN	X	XOR
G	GOTO	P	PRINT	Y	Boş
H	HEX\$	Q	Boş	Z	Boş
I	INPUT	R	RUN		



Tab: Tabülatör tuşuna her basışta imleç **8 karakter** sağa kayar.



NumLock: Sayısal klavyeyi kontrol modundan çıkartarak, hesap moduna çevirir. Bu tuşun üzerinde ışıklı bir gösterge vardır. Bastığımızda bu ışıklı gösterge yanar. Hesap moduna geçilir. Bir kez daha basarsak ışıklı gösterge söner ve kontrol moduna geçilir.



Ctrl tuşu diğer tuşlarla birlikte kullanılır. ^ işareti ile tanımlanır. ^C, ^H gibi... Diğer tuşlarla birlikte görevleri aşağıda verilmiştir:

<u>TUŞ</u>	<u>GÖREVİ</u>
^B	Bir önceki sözcüğe geçilir.
^C	Programın çalışmasını durdurur.
^E	Satır sonuna kadar siler.
^F	Bir sonraki sözcüğe geçilir.
^G	Bip sesi verir.
^H	Bir harf siler.
^I	Tab yapar. (8 karakter ileri)
^K	İmleci sayfa başına getirir.
^L	Ekranı temizler.
^N	İmleci satır sonuna götürür.
^R	Satır içinde, araya karakter sokma moduna geçilir.
^T	Fonksiyon tuşlarının durumlarını ekranın altında gösterir.
^Y	Bulduğumuz satırdan bir önceki satırı gösterir.



Escape: Kullanılan programa göre değişiklik gösterir. Daha çok geri dönüş ve çıkışlar için programlanır.

SORULAR

1. Tam Sayı nedir?
2. Tek Duyarlıklık Sayı nedir?
3. Çift Duyarlıklık Sayı nedir?
4. Alt tuşunun işlevleri nelerdir?
5. Ctrl tuşunun işlevleri nelerdir? Hangi tuşlarla birlikte nasıl çalışır?
6. Fonksiyon tuşlarının genel amacı nedir?
7. BASIC'e geçerken kullanılan anahtarların görevleri nelerdir?
8. (,) virgül (;) noktalı virgül, (:) iki nokta işaretinin BASIC'deki anlamı nedir.
9. $16.02E+4$ üslü sayısının matematiksel karşılığı nedir?
10. İşlem öncelik sıralarını sayınız.

BÖLÜM X.

BASIC'İN TEMEL YAPISI

A. KOMUT TÜRLERİ

BASIC dilinde kullanılan komut türleri aşağıdadır:

1. Ekran silme, çalıştırma, açıklama ve sona erdirmeye komutları.
2. Atama ve Giriş/Çıkış komutları.
3. Ekran düzenleme komutları.
4. Sistem komutları.
5. Disk/Disket komutları.
6. Kontrol komutları.
7. Döngü komutları.
8. Dizinli değişken komutları.
9. Hazır fonksiyon komutları.
10. Dosya düzenleme komutları.

B. SATIR NUMARASI VERME

BASIC yorumlayıcısı iki modda çalışır:

1. Direkt Mod
2. Endirekt Mod

Direkt Mod: Bu modda çalışırken, klavye yardımıyla komut ve istenenler girilir ve ENTER'e bastığımızda sonuç hemen karşımıza çıkar.

Örnek:

```
PRINT 20*5 ↵          PRINT "BILGISAYAR" ↵
100                   BILGISAYAR      gibi.
```

Ne var ki, bir program, birden çok satırla oluşturulur. Bunun için de bir sıra takip etmek gerekir. Bu nedenle program yazımı satır numaraları ile yapılır.

Endirekt Mod: Satır numaralarıyla yapılan programlamadır. Yukarıda da belirttiğimiz gibi, bundan sonra bütün programlarımızı satır numaraları vererek yapacağız.

Satır numaraları **0-65529** arasındaki sayılardan oluşur. **1,2,3,.....n** şeklinde gideceği gibi, **100,200,300,400,.....n** veya **10,23,25,30,34,.....n** şeklinde de gidebilir. Ancak biz satır numaralarını **10**'dan baslatıp **10**'ar artırarak yürütmeyi alışkanlık haline getireceğiz. Bunu yapmamızın nedeni, programın ileri bir aşamasında aralara satır eklemek içindir. Önemli olan satırların artarak değer almasıdır. Bir düzen içinde gitmesi önemli değildir. Bir kez daha yinelemekte yarar var; satır numaralarını **10**'ar artırarak yürüteceğiz.

C. BİR ALT SATIRA GEÇME

Bir satıra en çok **255** karakter yazı yazabiliriz. Yazmak istediğimiz komut ya da işlemleri yazdıktan sonra, bir alt satıra geçmek veya belleğe göndermek için mutlaka **ENTER** tuşuna basmalıyız. Satır bittikten sonra, **ENTER**'e basılınca imleç bir alt satıra geçecek ve yazdığımız satırı belleğe gönderecektir.

D. SATIR EKLEME

Program yazımı sırasında veya bitirilip test edildikten sonra çalışmazsa, düzeltme yoluna gidilir. Bazen öyle bir durum olur ki, satır üzerindeki düzeltme yetmeyebilir. Bu durumda araya eksik olan yeri, yeni bir satırla ekleriz. Şimdi aşağıdaki programı bilgisayara yazalım.

Örnek:

```
10 CLS
20 REM DIKDÖRTGENİN ALANINI HESAPLAYAN PROGRAM
30 INPUT "KISA KENARI GIRINIZ";KK
40 INPUT "UZUN KENARI GIRINIZ";UK
50 PRINT "DIKDÖRTGENİN ALANI";ALAN;"METREKAREDİR"
60 END
```

(Satır sonlarında bir alt satıra geçmek ve satırı belleğe göndermek için **ENTER** tuşuna basınız.)

Bu programı yazıp çalıştırdıktan sonra hangi verileri girerseniz girin, sonuç 0 çıkacaktır. Çünkü programda yanlışlık, -daha doğrusu eksiklik- vardır. Biz bu programda alan hesaplamasını yapmayı unuttuk. Şimdi bu eksikliğini ekleyelim.

Bu durumda hemen alt kısımda bir yerde, girmek istediğimiz satırların arasında, yeni bir satır numarası verilir. Bu programda alan hesabını **40.** ve **50.** satırların arasına eklememiz gerekmektedir.

```
45 ALAN=KK*UK
```

Yeniden liste istediğimizde **45** numaralı satırın **40.** ve **50.** satırlar arasında yerini aldığını görürüz. Artık çalıştırdığımızda istediğimiz sonucu verecektir.

E. SATIR SİLME

DELETE

Program yapımı sırasında bazen gereksiz yazılan satırlar da olabilir. Bu satırları **DELETE** komutuyla sileriz. **Direkt Modda** yazılır.

DELETE 50	↵	Yalnız 50. satır silinir.
DELETE -80	↵	Baştan 80. satıra kadar silinir.
DELETE 100-140	↵	100 ve 140. satırlar arası silinir.
DELETE 170-	↵	170. satırdan sona kadar silinir.

NOT: ↵ işaretleri **ENTER** tuşuna basılacağını gösterir.

Yalnızca bir satır sileceksek, satır numarasını yazdıktan sonra **ENTER**'le boş geçerseniz o satır silinir.

80 ↵ **80.** satır silinir.

F. PROGRAMI LİSTELEME

LIST

Bir programın içeriğini (Listesini) görmek istediğimiz zaman **LIST** komutu kullanılır. Bu komut bu durumda, **Direkt Mod**'da yazılır.

LIST komutu: Bir programın listesini görmek için kullanılır. Çeşitli kullanım biçimlerini görelim:

LIST	↵	Programın tamamını listeler.
LIST 100	↵	Yalnızca 100. satırı gösterir.
LIST -80	↵	Programın başından 80. satıra kadar listeler.
LIST 80-120	↵	80. ve 120. satırlar da dahil arasını listeler.
LIST 120-	↵	120. satırdan sonuna kadar listeler.

Programın listesini Yazıcı'dan almak istersek **LIST** komutunun başına **L** harfini yazarak, **LLIST** şeklinde yukarıdaki kurallar doğrultusunda listeyi yazıcıdan alabiliriz.

Ya da **LIST** ile birlikte **LPT**'yi kullanarak yazıcıdan listeyi yine alabiliriz.

LIST 80-120 "LPT:" gibi.

G. PROGRAMI DİSKE KAYIT ETME

SAVE

Programı yazdıktan sonra diskete kayıt ederiz. Bunu **SAVE** komutuyla yaparız. Yazdığımız programa, **8** karakteri geçmeyen bir ad vererek kayıt edebiliriz. **F4** tuşuna bu komut yüklenmiştir.

Yazılışı:

SAVE <"Dosya Adı">, [, A, P]

SAVE "BORDRO"	BORDRO.BAS kayıt edilir.
SAVE "B : BORDRO"	B sürücüsüne kayıt edilir.
SAVE "BORDRO" ,A	ASCII kodda kayıt edilir.
SAVE "BORDRO" ,P	Korumalı (Protect) kayıt edilir. Program kilitletir. Bu şekilde kayıt ettiğimizde programın listesini bir daha alamayız.

SAVE "C : \BASIC\BORDRO" C sürücüsünde bulunan **BASIC** dizinine **BORDRO** adlı programı kayıt eder.

H. YENİ BİR PROGRAM YAPMAYA GEÇME

NEW

Bir programı bitirdikten veya disketten çağırdığımız programla işimiz bittikten sonra yeni bir program yapacaksa, belleği mutlaka temizlememiz gerekir. Eğer belleği temizlemezsek, eski programla yeni programın satır numaraları birbirine karışır. Eski programda olup da yeni programda, olmayan satır numaraları varsa, bunlar yeni programa geçer.

Bu nedenlerle, her yeni program yapmaya geçişte belleği **NEW** komutuyla temizlememiz gerekir. Anlamı yenidir. Direkt modda yazmalıyız.

NEW ↵
Ok

Şeklinde yazdığımızda, **Ok** mesajıyla belleğin temizlendiğini bildirir. Artık yeni bir program yapmaya geçebiliriz.

I. DİSKETTEN PROGRAM ÇAĞIRMA

LOAD

Bir programı belleğe yüklemek için **LOAD** komutunu kullanırız. Bunu da direkt moda yaparız. **LOAD** komutundan sonra "... " içinde, çağırmak istediğimiz programın adını yazarız ve **ENTER**'e basarız.

Yazılışı:

LOAD "Dosya Adı" [,R]

LOAD "BORDRO"

LOAD "BORDRO"

LOAD "BORDRO",R

LOAD "B:BORDRO"

LOAD "C:\BASIC\BORDRO"

BORDRO adlı programı disketten belleğe yükler (Çağırır.)

BORDRO adlı programı disketten çağırır ve program otomatik olarak çalışır.

B sürücüsündeki **BORDRO** adlı programı çağırır.

C Sürücüsünde, **BASIC** dizini içindeki **BORDRO** adlı programı çağırır.

LOAD "Dosya Adı" ↵

Eğer programımızı kaydederken uzantı vermeden kaydetmiş isek, otomatik olarak, **BASIC** programı olduğu için **.BAS** uzantısını alacaktır. Bu durumda **LOAD** komutu ile programı çağırırken uzantısını yazmaya gerek yoktur. Ama kaydederken değişik bir uzantı vermiş isek uzantıyı yazmamız gerekir.

UCRET.BAS ve **VERGLIAD** adlı programları disketten çağıralım:

LOAD "UCRET" ↵

LOAD "VERGI.IAD" ↵

Program belleğe yüklendikten sonra, karşımıza **Ok** şeklinde, Tamam anlamında mesaj çıkar. Bu Okey sözcüğünün kısaltılmışıdır. Bu durumda programımız bellekte demektir. Eğer, bu adla bir program diskette yoksa Bad file name mesajı çıkararak, böyle bir dosya olmadığını belirtecektir.

Programı belleğe aldıktan sonra, direkt mod'da **RUN** komutu ile çalıştırabiliriz.

J. PROGRAMI ÇALIŞTIRMA

RUN

Yaptığımız bir programı çalıştırmak için **RUN** komutu kullanılır. Bir programı çalıştırabilmek için o programın bellekte bulunması gerekir. Eğer programı yeni yaptıysak, program bellektedir. Ama daha önce yapıp diskete kaydettiğimiz bir programı çalıştırabilmek için o programı belleğe yüklememiz gerekir.

K. DOSYA LİSTESİNİ GÖRME

FILES

Basic'le çalışırken, diskette kayıtlı dosyaların listesini görmek istersek bunu **FILES** komutuyla yaparız. Bu komut, **MS-DOS**'daki **DIR** komutunun karşılığıdır. Direkt moda yazılır.

FILES	↳	Bulduğumuz sürücüdeki dosyaların adlarını verir.
FILES "B:"	↳	B sürücüsündeki dosyaların adlarını listeler.
FILES "*.BAS"	↳	BAS uzantılı dosyaları listeler.
FILES "M*.*"	↳	M harfi ile başlayan dosyaların listesini verir.
FILES "C:\BASIC\"	↳	C sürücüsündeki BASIC dizininde bulunan dosyaları listeler.

L. DOSYA ADI DEĞİŞTİRME

NAME ... AS

Diskete kaydettiğimiz herhangi bir doyanın adını değiştirmek istediğimizde **NAME** "**Eski Ad.Uzantı**" **AS** "**Yeni Ad.Uzantı**" şeklinde yazarak programın adını değiştirebiliriz. Daha önce diskette **ORNEK.BAS** adı ile kaydettiğimiz programın adını **ORNEK2.BAS** olarak değiştirelim:

```
NAME "ORNEK.BAS" AS "ORNEK2.BAS" ↵
```

FILES komutu ile disketteki dosyaların adını yeniden görmek istersek **ORNEK.BAS** programının **ORNEK2.BAS** olarak değiştiğini görürüz.

M. DİSKETTEN DOSYA SİLME

KILL

Basic ortamında çalışırken disketten herhangi bir dosyayı silmek istediğimizde **KILL** (Öldür) komutunu kullanırız. Bir örnek yaparak bunun çalışmasını görelim:

Daha önce diskete kaydettiğimiz **DENEME** adlı programı silelim. Biz bu programa anımsayacağımız üzere uzantı vermemiştik. Bu nedenle bu programın uzantısı **.BAS** oldu. Bunu görmek için **FILES** komutunu kullanabiliriz.

```
KILL "DENEME . BAS" ↵
```

Tekrar **FILES** yaparsak, **DENEME** adlı programın disketten silindiğini görürüz.

N. BASIC ORTAMINDAN DOS'A DÖNÜŞ (ÇIKIŞ) SYSTEM

Basic ortamında çalışmamız bittikten sonra ya da herhangi bir nedenle **DOS'a** dönmek istersek **SYSTEM** komutu ile döneriz.

```
SYSTEM ↵
```

Yaptıktan sonra karşımıza işletim sisteminin promptu çıkar.

SORULAR

1. Komut türlerini sayınız.
2. Direkt ve endirekt mod nedir?
3. Satır silme yöntemleri nelerdir?
4. Araya satır ekleme nasıl yapılır?
5. BASIC'de dosya listesini nasıl görürüz?
6. Yeni bir program yapmaya nasıl geçilir?
7. Dosya adı nasıl değiştirilir?
8. Program disketten nasıl çağrılır?
9. Program nasıl çalıştırılır?
10. Program diskete nasıl kaydedilir?
11. BASIC'ten çıkıp DOS'a nasıl dönülür?

BÖLÜM XI.

BASIC PROGRAMLAMA DİLİ

KOMUT ve DEYİMLERİ

Program yapmaya geçmeden önce, komutlar **Türkçe** olsaydı bir programı nasıl yapardık? Sorusunun cevabını arayalım, sonra da program yapmak için gerekli komutları görelim:

Komutların Türkçe olduğunu varsayarak iki sayıyı toplayan programı yapalım:

```
10 EKRANI TEMİZLE
20 GIR "BIRINCI SAYI";A
30 GIR "IKINCI SAYI";B
40 HESAPLA TOPLAM=A+B
50 YAZ "IKI SAYININ TOPLAMI";TOPLAM;" `DIR"
60 BITIR
```

Şimdi aynı programı BASIC diliyle kodlayalım ve benzerliğini görelim:

```
10 CLS
20 INPUT "BIRINCI SAYI";A
30 INPUT "IKINCI SAYI";B
40 LET TOPLAM=A+B
50 PRINT "IKI SAYININ TOPLAMI";TOPLAM;" `DIR"
60 END
```

A. EKRAN SİLME, ÇALIŞTIRMA, AÇIKLAMA VE SONA ERDİRME KOMUTLARI

1. EKRANI SİLME

CLS

Yazılışı:
CLS

CLS komutu ekranı silmeye yarar. Bu komutu kullandığımızda ekrandaki görüntüler silinir. Yalnız bellekteki bilgilerin silinmeyeceğini unutmayalım; yalnızca ekran temizlenecektir. **CLS**'nin kullanıldığı satırda ekrandaki görüntüler silinerek, ekranı istediğimiz biçimde kullanabileceğiz; bundan sonra ekranda yeni verdiğimiz görüntüler yeracaktır. (Bir kez daha yinelemekte yarar var: Ekrandaki görüntüler silindiği halde, bellekteki bilgiler silinmez.)

2. PROGRAMI ÇALIŞTIRMA

RUN

Yaptığımız bir programı çalıştırmak için **RUN** komutu kullanılır. Bir programı çalıştırabilmek için o programın bellekte bulunması gerekir. Eğer programı yeni yaptıysak, program bellektedir. Ama daha önce yapıp diskete kaydettiğimiz bir programı çalıştırabilmek için o programı belleğe yüklememiz gerekir.

3. AÇIKLAMA YAZMA

REM

Program satırlarına açıklama yazmak için kullanılır. **REM** satırları programın çalışması sırasında, çalışmayı etkilemez ve işlem görmezler. Programın listesi alındığında bilgi olması amacıyla görünürler. **REM** yerine ' (Tek Tırnak) işareti de **REM** deyiminin görevini üstlenir.

```
10 REM FAIZI HESAPLAYAN PROGRAM
20 REM HAZIRLAYAN: GÜVEN TANIŞ
30 INPUT "KAPITALI GIRINIZ";K
40 INPUT "SÜREYİ GIRINIZ";N
50 INPUT "FAIZ FİYATINI GIRINIZ";T
60 ' K İLE KAPITAL, N İLE SÜRE, F İLE FAIZ FİYATI TANIMLANMIŞ
TIR.
70 F=(K*N*T)/1200 : ' FAIZ HESAPLANIYOR.
80 PRINT "PARANIZIN FAIZI";F;"LİRADIR" ' SONUÇ YAZDIRILYOR
```

Yukarıdaki programda **REM** komutunun birkaç yazılış biçimini görüyorsunuz. Programı çalıştırdığımızda **30** nolu satırdan çalışmaya başlayacaktır. **60** nolu satır **REM** satırı olduğu için çalışma sırasında yine gözönüne alınmayacaktır. **70. 80.** ve **90.** satırlardan sonra ' işareti yine açıklama yazmak için **REM** yerine kullanıldığından çalışmayı etkilemeyecektir.

4. PROGRAMI SONA ERDİRME

END

Yazılışı:
END

Şimdiye kadar yaptığımız programlarımızı çalıştırdığımızda, program sona erdikten sonra **Ok** mesajı yerine **Break in X** (X son satırın numarası) mesajıyla karşılaştık. Program sonlarında bu nedenle **END** komutunu kullanarak programı bitireceğiz.

```
10 PRINT "BILGISAYAR"  
20 PRINT "ÖĞRENIYORUM"  
30 END
```

Artık çalıştırdığımızda **Ok** mesajı çıkacaktır.

```
RUN ↵  
BILGISAYAR  
ÖĞRENIYORUM  
Ok
```

5. PROGRAMI DURDURMA STOP

Programın çalışmasını durdurur. Çalışma sırasında **STOP** yazan satıra geldiğinde program durur ve kaçınıcı satırda durduğunu belirtir. **Break in 120** şeklinde, **120.** satırdayım anlamında mesaj verir. Programın kaldığı yerden devam etmesi için **CONT** deyimini veya tuşu kullanılır.

```
10 CLS  
20 PRINT "EN BÜYÜK KIM?"  
30 PRINT "Öğrenmek için F5 tuşuna basınız veya CONT yazınız."  
40 STOP  
50 PRINT "GALATASARAY"  
60 END
```

Programı çalıştıralım:

Ekran temizlenir ve;

```
RUN ↵  
EN BÜYÜK KIM?  
Öğrenmek için F5 tuşuna basınız veya CONT yazınız.  
Break in 40
```

Yazar ve bekler. Devam etmesi için **CONT** yazalım veya **F5** tuşuna basalım.

```
GALATASARAY  
Ok
```

Yazarak program sona erer. Bu programı basit bir şaka gibi arkadaşlarınıza çalıştırtabilirsiniz.

B. ATAMA ve GİRİŞ/ÇIKIŞ KOMUTLARI

1. EKRANA ve YAZICIYA YAZDIRMA

a. PRINT

Yazılışı:
PRINT "Mesaj"; [Değişken Adı]
ya da
PRINT [Değişken Adı]

Ekranda görüntülenmesini istediğimiz mesaj ya da değişkenleri yazmada kullanılır.

```
PRINT "BILGISAYAR"  ↵  
BILGISAYAR  
Ok
```

Yukarıda **PRINT** komutuyla **Ekrana Yaz** anlamında komut verdik. **BİLGİSAYAR** sözcüğü Alfasayısal olduğu için **"..."** (Çift Tırnak) içerisinde yazdık. Bilgisayar, bizim verdiğimiz komutu yerine getirerek ekrana **BİLGİSAYAR** sözcüğünü yazdı ve **TAMAM** anlamında **Ok** (Okey) yazarak işlemi yerine getirdiğini bildirdi.

```
PRINT 3584  ↵          PRINT (25*6)/2  ↵  
3584          75  
Ok           Ok
```

Sayısal değerlerin yazılışında **"** işareti kullanılmaz. Birinci örneğimizde **3584** sayısını, ikinci örneğimizde **25**'le **6**'yı çarpmasını ve **2**'ye bölmesini istedik bunu yerine getirdi.

Burada dikkat edeceğimiz önemli bir konu da sayıların, bir karakter içeriden yazılmasıdır. Bunun nedeni, sayının (-) olması durumunda bu işaretin görüntülenmesi içindir. Aynen matematikte olduğu gibi; sayı (+) ise önüne hiçbir işaret konmaz.

Burada kullandığımız yazım şekline **Direkt Mod** adını veriyoruz.

Şimdi **PRINT** komutu ile kullanılan işaretlerin, hangi durumlarda nasıl kullanılıp yazıldığını görelim.

```
PRINT "MERHABA! BILGISAYAR DERSINE HOŞ GELDİNİZ."  ↵  
MERHABA! BILGISAYAR DERSINE HOŞ GELDİNİZ.
```

Bir sözcük grubunu (**String, Katar**) yazmak istediğimizde, çıkmasını istediğimiz şekilde "... " içinde **PRINT** komutunun yanına yazarız. Yukarıdaki örnekte bu yazım biçimi gösterilmiştir. Dikkat ettiyseniz, sözcükler arasındaki boşlukları da string ifadeye dahil ettik.

Aşağıda vereceğimiz örneklerdeki yazılımlar bize yazmak istediğimiz değer ve ifadelerin nasıl yazıldığını gösterecektir.

```
PRINT 15;20;-35;40;120;210  ↵
15 20-35 40 120 210
Ok
```

Daha önce de belirttiğimiz gibi sayıları "." içinde yazmıyoruz. Burada aralarına ; (Noktalı Virgül) koyarak, sayıların yanyana yazılması için komut verdik. İstediklerimiz sayı olduğu için - ya da + olacağı düşünülerek aralarında birer karakterlik boşluk verdi.

```
PRINT "YANYANA";"YAZMAYA";"ÖRNEKTİR"  ↵
YANYANAYAZMAYAÖRNEKTİR
Ok
```

Burada da alfasayısal ifadelerin yanyana yazılmasını istediğimizden, sözcükler arasında hiç ara vermeden yazdı. Bu sözcüklerin böyle bir yazılımda nasıl ayrılacağını ileride göreceğiz.

```
PRINT "ARALIKLI","YAZMAK","İÇİN","ÖRNEKTİR"  ↵
ARALIKLI      YAZMAK      İÇİN      ÖRNEKTİR
Ok
```

Burada ifadeler arasına, yani "... " içinde yazılan alfasayısal ifadeler arasına , (Virgül) koyarak aralıklı yazmasını istedik. Bu sözcükler arasında 16'şar karakterlik bir ara vardır.

```
PRINT 15,20,130,210  ↵
15          20          130          210
Ok
```

Bu örneğimizde de sayıları aralıklı yazmasını istedik. Yine 16'şar karakterlik ara vererek yazdı.

```
PRINT "EREĞLI","KONYA"  <Enter>
EREĞLI      KONYA
Ok
```

Yine bu örneğimizde **EREĞLİ** ve **KONYA** sözcüklerinin aralarına , (Virgül) işaretini koyduk ve aralıklı yazdı. Aşağıda vereceğimiz örnekte de virgülün yerini değiştirelim, sonucu görelim:

```
PRINT "EREĞLI, KONYA" ↵  
EREĞLI, KONYA  
Ok
```

Dikkat ettiyseniz Virgül'ü burada "... " içine yazdık. Bu nedenle çıktıda virgül işareti de mesaja dahil olduğu için, virgülü de yazdı.

İki örneğimizi inceleyecek olursak, birincisinde virgül işaretini "... " dışına yazdığımızdan **aralıklı yaz** anlamında komut durumundaydı. İkincisinde ise virgül'ü "...." içerisinde yazarak virgülün de alfasayısal ifadeye dahil olduğunu belirttik ve onu da yazmasını istedik. Bu nedenle virgül işaretini de çıktıda yazmış oldu.

```
PRINT "19 MAYIS 1919" ↵  
19 MAYIS 1919  
Ok
```

Burada hem sayısal hem de alfabetik değer ve ifadeleri yazmasını istedik. Bunların ikisini de "... " içinde yazdığımızdan alfasayısal bir ifade olduğu için çıktısı "...." içinde yazdığımız şekilde oldu.

Aynı yazıyı bu kez de başka biçimde yazalım ve aralarındaki farkı görelim.

```
PRINT 19;"MAYIS";1919 ↵  
19 MAYIS 1919  
Ok
```

Burada da sayısal değerleri "... " dışında yazdık. Alfasayısal ifadeleri "... " içinde yazdık. Ama aralarına ; (Noktalı Virgül) koyarak yanyana yazmasını istedik.

Buraya kadar yaptığımız örnekler **Direkt Mod**'da yapılmıştı. Bu yöntemle **PRINT** komutunun çeşitli yazılış ve kullanılış biçimlerini gördük.

Bundan sonra Endirekt Mod dediğimiz, satırlara numara vererek yazmayı görelim. Çünkü bir program birden çok satırdan oluşur. Bilgisayar yaptığımız programı çalıştırdığımızda her satırı tek tek okuyarak, verilen emirleri yerine getirir.

Yukarıda yaptığımız bütün örnekleri bu kez de, satır numaraları vererek tek bir program içerisinde toplayalım ve sonucunu görelim:


```
10 PRINT 15;20;-35;40;120;58
20 PRINT "YANYANA";"YAZMAYA";"ÖRNEKTİR"
30 PRINT "ARALIKLI","YAZMAK","İÇİN","ÖRNEKTİR"
40 PRINT 15,20,-60,120
50 PRINT "19 MAYIS 1919"
60 PRINT 19;"MAYIS";1919
70 PRINT "EREĞLİ","KONYA"
80 PRINT "EREĞLİ,KONYA"
90 PRINT "BU SATIR NORMAL YAZIM İÇİN ÖRNEKTİR"
```

NOT: Programı yazarken, bir alt satıra geçmek veya satırı belleğe göndermek için ↵ (ENTER veya RETURN) tuşuna mutlaka basınız.

Bu programı çalıştırdığımızda:

```
RUN ↵
 15 20-35 40 120 58
YANYANAYAZMAYAÖRNEKTİR
ARALIKLI          YAZMAK          İÇİN          ÖRNEKTİR
 15              20              -60              120
19 MAYIS 1919
 19 MAYIS 1919
EREĞLİ          KONYA
EREĞLİ,KONYA
BU SATIR NORMAL YAZIM İÇİN ÖRNEKTİR"
```

Bu kez de değişik satırlarda yazdığımız halde çıktısı yanyana ve aralıklı olan bir örnek program yapalım.

```
10 PRINT "GÜVEN" ,
20 PRINT "TANIŞ"
```

Programı çalıştıralım:

```
RUN ↵
GÜVEN          TANIŞ
Ok
```

Bunu nasıl yaptığımızı açıklayalım: Dikkat ettiyseniz **10.** satırın sonunda , (virgül) işaretini yazdık. Bunun anlamı, bundan sonra gelecek yazımı aralıklı olarak yaz olduğundan virgülden sonra gelen mesajı aralıklı olarak yazdı.

PRINT komutundan sonra hiçbir şey yazmazsak, bir satır boş bırakır. Aynen dakti-
loda bir satır ara vermek için bir kol attığımız gibi, bir satır boş bırakacaktır. Boş geçilmesi
istenilen satır kadar **PRINT** komutu kullanılır.

```
10 PRINT "1. SATIR"  
20 PRINT "2. SATIR"  
30 PRINT "3. SATIR"  
40 PRINT  
50 PRINT  
60 PRINT  
70 PRINT "7. SATIR"
```

Programı çalıştıralım:

```
RUN ↵  
1. SATIR  
2. SATIR  
3. SATIR  
  
7. SATIR
```

Gördüğümüz gibi **1. 2. ve 3.** satırdan sonra **3 Satır** boş bırakarak **7. SATIR**'ı yazdı.

PRINT komutunun yerine ? Soru İşareti'ni de kullanabiliriz.

```
10 ? "FIKRET"  
20 ? "TANIŞ" gibi.
```

PRINT komutu bu şekliyle de istediğimiz bilgileri ekrana verir. Eğer tekrar **LIST**
yapacak olursak ? işaretlerinin yerine **PRINT** geldiğini görürüz.

Örnek: Ekranın ortasına **CLS**'yi de kullanarak **MUSTAFA KEMAL ATATÜRK**
yazdıralım:

Ekranın **24** satır, **80** karakterden (Sütun) oluştuğunu düşünürsek, ekranın ortası **12.**
satırdır. Buna göre:

```
5 CLS
10 PRINT
20 PRINT
30 PRINT
40 PRINT
"
"
"
"
120 PRINT "                MUSTAFA KEMAL ATATÜRK"
```

Programı çalıştırdığımızda 11 satır boşluk bırakarak, 12. satırda çift tırnaktan sonraki boşlukları da gözönüne alarak MUSTAFA KEMAL ATATÜRK ekranın ortasında yer alacaktır.

b. LPRINT

Bilgisayarda, yazdığımız bilgileri ekran yerine yazıcıdan almak istediğimizde kullanılır. Girdiğimiz verileri ve sonuçları istediğimiz formatta kağıda bu komutla aktarabiliriz. PRINT yerine LPRINT yazdığımız satırların yazıcıdan alınmasını sağlamış oluruz.

Yazılışı:
LPRINT <Kod>; "Mesaj"; <Değişken>, <Değişken>, . . .

```
10 LPRINT "YAZICIDAN ÇIKTI ALMAYA ÖRNEKTİR."
```

Bu mesajımız ekran yerine yazıcıdan çıkacaktır.

Aşağıda değişik karakterler yazdırabilmek için parametler verilmiştir:

<u>Karakter Kodu</u>	<u>Görevi</u>
CHR\$(27)+"W1"	Büyütülmüş yazı
CHR\$(27)+"W0"	Büyütülmüş yazı iptali
CHR\$(27)+CHR\$(15)	Küçültülmüş yazı
CHR\$(27)+CHR\$(18)	Normal yazı
CHR\$(27)+"4"	İtalik yazı
CHR\$(27)+"5"	İtalik yazı iptali
CHR\$(27)+"E"	Koyu yazı
CHR\$(27)+"F"	Koyu yazı iptali
CHR\$(27)+"-1"	Alt çizgili yazı
CHR\$(27)+"-0"	Alt çizgi iptali
CHR\$(27)+"W" ; CHR\$(1)	Geniş yazı
CHR\$(27)+"W" ; CHR\$(0)	Geniş yazı iptali
CHR\$(27)+"G"	Çift baskılı yazı

CHR\$(27)+"F"	Çift baskı iptali
CHR\$(27)+"S";CHR\$(1)	Üst indis
CHR\$(27)+"T";CHR\$(0)	Üst indis iptali
CHR\$(27)+"S";CHR\$(0)	Alt indis
CHR\$(27)+"T";CHR\$(0)	Alt indis iptali

```
10 CLS
20 W$=CHR$(27)
30 LPRINT W$+"W1";"Büyütülmüş yazı"
40 LPRINT W$+"W0";"Büyütülmüş yazı iptali"
50 LPRINT W$+CHR$(15);"Küçültülmüş yazı"
60 LPRINT W$+CHR$(18);"Normal yazı"
70 LPRINT W$+"4";"Italik yazı"
80 LPRINT W$+"5";"Italik yazı iptali"
90 LPRINT W$+"E";"Koyu yazı"
100 LPRINT W$+"F";"Koyu yazı iptali"
110 LPRINT W$+"-1";"Alt çizgili yazı"
120 LPRINT W$+"-0";"Alt çizgi iptali"
130 LPRINT W$+"W";CHR$(1);"Geniş yazı"
140 LPRINT W$+"W";CHR$(0);"Geniş yazı iptali"
150 LPRINT W$+"G";"Çift baskılı yazı"
160 LPRINT W$+"F";"Çift baskı iptali"
170 LPRINT W$+"S";CHR$(1);"Üst indis"
180 LPRINT W$+"T";CHR$(0);"Üst indis iptali"
190 LPRINT W$+"S";CHR$(0);"Alt indis"
200 LPRINT W$+"T";CHR$(0);"Alt indis iptali"
210 END
```

c. WRITE

Ekranı mesaj yazdırmak için kullanılan bir başka komuttur. **PRINT**'ten farkı mesajları " (Çift tırnak) içerisinde yazdırır ve aralarına , (Virgül) koyar.

Yazılışı:
WRITE "mesaj" ;Değişken

Aşağıdaki örneğimizde **PRINT** ve **WRITE** birlikte kullanılmıştır. Aralarındaki farkı görebilirsiniz:

```
10 PRINT "GÜVEN", "TANIŞ"  
20 WRITE "GÜVEN", "TANIŞ"  
30 PRINT "9 MART 1961"  
40 WRITE "9 MART 1961"  
50 PRINT "KONYA, EREĞLİ"; "TÜRKİYE"  
60 WRITE "KONYA, EREĞLİ"; "TÜRKİYE"
```

```
RUN ↵  
GÜVEN          TANIŞ  
"GÜVEN", "TANIŞ"  
9 MART 1961  
"9 MART 1961"  
KONYA, EREĞLİTÜRKİYE  
"KONYA, EREĞLİ", "TÜRKİYE"  
Ok
```

2. DEĞER YA DA İFADE ATAMA

LET

Yazılışı:
[LET] <Değişken>=<İfade>

Bilgileri değişkenlere atamak için kullanılır. **LET** deyimi eşitliğin sağındaki ifadeyi, solundaki değişkene (adrese) depolar.

```
LET A = 100
```

100 sayısının A'ya depolanması içindir. Bundan sonra A'nın değeri **100** olacaktır.

```
LET AD$="FİKRET"
```

FİKRET sözcüğünü **ADS** alfasayısal değişkenine depolar. **ADS** bundan böyle **FİKRET** olur.

```
10 LET A=50
20 LET B=70
30 LET C=A+B
```

Burada A'ya **50**, B'ye **70** değerini verdik. C'ye ise **A+B** değerini yani, **A** ve **B** değerinin toplamını depolamış olduk. Böylece C'nin değeri **120** olacaktır. Bu programı çalıştırdığımızda ekranda herhangi bir şey göremeyiz. Eğer sonucu görmek istersek yeni bir satır ekleyerek ekrana yazmasını isteriz.

```
40 PRINT C
```

Satırını ekleyerek sonucu görmüş oluruz. Ekranda **120** sayısı görülecektir.

Alfasayısal ifadeleri depolamak istediğimizde, yine **PRINT**'te olduğu gibi "..."
içerisinde yazıyoruz.

```
10 LET A$="BİLGİSAYAR"
20 LET B$="DERSİNİ"
30 LET C$="SEVİYORUM"
40 LET D$=A$+B$+C$
50 PRINT D$
```

Programı çalıştırdığımızda:

```
RUN
BİLGİSAYARDERSİNİSEVİYORUM
```

Çıktısını alırsınız. Burada (+) işareti ata anlamındadır. Matematiksel toplama anlamında değildir.

Programda **A\$, B\$, C\$** değişkenlerine teker teker **BİLGİSAYAR, DERSİNİ, SEVİYORUM** sözcükleri depolanmış, bunlar da **D\$**'a aktarılmıştır.

LET DEYİMİNİN ÖZELLİKLERİ

1) Sayısal değerler, sayısal değişkenlere; alfasayısal ifadeler, alfasayısal değişkenlere atanmalıdır.

```
LET A=100          LET A$="KAMURAN"   gibi.
```

2) Eşitliğin sol tarafındaki değişken, son atanan değeri alır.

```
10 LET A=5
20 LET B=8
30 LET A=A*B/2
.
.
.
130 LET C=A+B
```

Burada 30. satırdan sonra A'nın değeri 20'dir. 130. satıra kadar da böyle gidecektir. İlk satırda A=5 değerini aldı. 30. satırda A'ya yeni bir değer verildi. 130. satırda C'nin değeri 20+8=28 olacaktır.

3) Eşitliğin solunda hiçbir matematiksel işlem yapılmaz.

```
LET A+B=C yazılması yanlıştır. LET C=A+B olmalıdır.
```

4) Eşitliğin sağ tarafında değişken kullanılıyorsa, daha önceden mutlaka bu değişkenlere değer atanması gerekir.

```
10 LET A=B+C yazımı, ilk satır veya daha önce değer verilmemiş ise yanlıştır.
```

```
10 LET B=25
20 LET C=30
30 LET A=B+C   olmalıdır.
```

5) LET yazmadan değişkenlere değer atanabilir.

```
10 LET A=20           yerine   10 A=20
20 LET A$="BELMA"    yerine   20 A$="BELMA"   gibi.
```

Buraya kadar gördüğümüz iki komutla basit programlar yapalım.

50000 TL'nin %50'den 6 ay sonra getireceği faizi hesaplayan ve sonucu yazan programı yapalım:

```
10 LET K=50000
20 LET N=6
30 LET T=50
40 LET F=(K*N*T)/1200
50 PRINT F
```

Şimdi yaptığımız örneği inceleyecek olursak, insan zekası ile bilgisayarın aynı yolu izlediğini görürüz. Bu yorum bundan sonra yapacağımız bütün programlar için geçerlidir. İnsan zekasının bir problemi çözmek için izlediği yolu burada da uygulayacağız.

10, 20 ve 30 no'lu satırlarda değişkenlere verilen değerleri atadık. 40 no'lu satırda problemin çözümü için gerekli formülü yazarak, işlemini elde etmiş olduk. Dikkat ederseniz burada F değişkeni "... " dışındadır. Bu nedenle oraya F'nin aldığı değer yazılacaktır.

Örnek: Kenarı 50, yüksekliği 30 cm olan üçgenin alanını hesaplayan programı yapalım:

```
10 LET A=50
20 LET H=30
30 LET ALAN=A*H/2
40 PRINT "ÜÇGENİN ALANI";ALAN;"SANTİMETRE KAREDİR"
```

Programı çalıştırdığımızda;

```
RUN ↵
ÜÇGENİN ALANI 750 SANTİMETRE KAREDİR      sonucu çıkacaktır.
```

3. KLAVYEDEN BİLGİ GİRME

INPUT

Yazılışı:
INPUT ["Açıklama Dizisi"]; <Değişken>

Örnek:
INPUT "ADINIZI SOYADINIZI GIRINIZ";ADŞ

Program içinde kullanılan değişkenlerin, değerlerinin klavyeden girilmesi için kullanılır. INPUT'tan hemen sonra ; (noktalı virgül) konursa girişten sonra aynı satırda kalmamızı sağlar. INPUT; "Mesaj";Değişken biçiminde. Değişkenden önce ; (virgül) yerine , (virgül) konursa soru işareti görüntülenmez. INPUT "Mesaj",Değişken biçiminde.

Şimdiye kadar yaptığımız programlarda, değişkenlerin değerlerini program içerisinden atadık. Böyle olunca, yeni bir problem çözebilmek için, yeniden program yapmak zorunda kalıyoruz. **INPUT** komutu bu sorunumuzu çözümlenecek, yaptığımız bir programla istediğimiz kadar problemi çözebileceğiz.

INPUT komutundan sonra, bir değişken adı belirlenir. Program çalışırken **INPUT** deyiminin bulunduğu satıra geldiğinde ? (Soru İşareti) belirir. Bunun anlamı klavye yardımı ile değişkene ilişkin bir ifade girilecek demektir. Programda belirtilen değişken sayısal ise, ? işaretinin yanına sayısal bir değer, alfasayısal ise alfasayısal bir ifade girilmelidir. Değişkenden önce ; yerine , işareti konursa ? işareti görüntülenmez.

```
10 INPUT A
20 INPUT ABŞ
```

Programı çalıştırdığımızda, karşımıza ? işareti çıkacaktır. İlk ? işaretinin karşısına bir sayısal değer girilmelidir. Sonra girdiğimiz bilgiyi belleğe göndermek için **ENTER** tuşuna basarız. Bundan sonra karşımıza ikinci bir ? işareti çıkacaktır. Bu satırda değişken için \$ işareti kullandığımızdan, alfasayısal bir değişken olduğu için alfasayısal bir ifade girmemiz gerekmektedir.

Şimdi Üçgenin Alanını hesaplayan programı **INPUT** ile yazalım ve çözelim:

```
10 INPUT A
20 INPUT H
30 LET ALAN=A*H/2
40 PRINT ALAN
50 END
```

Programı çalıştırdığımızda, karşımıza ? işareti çıkacaktır. Bizden kenarı girmemizi istemektedir. **60** sayısını girelim ve **ENTER**'e basalım. Yeniden soru işareti çıkar. Şimdi de yüksekliği **40** olarak girelim ve **ENTER**'e basalım. **120** sayısını sonuç olarak görürüz. Aşağıda iki örnek daha yapalım:

RUN	↵	RUN	↵
? 20	↵	? 50	↵
? 10	↵	? 20	↵
100		500	
Ok		Ok	

Sonuçları ile karşılaştırız. Görüldüğü gibi **INPUT** ile yaptığımız programlarla istediğimiz kadar soruyu çözebiliyoruz.

Yukarıda yaptığımız programlarda girilecek ifadelerin ne olduğu bilinmemektedir. Bu nedenle **INPUT** deyiminden sonra "..." içinde belirtilen değişkenlerin ne olduğunu, ne gireceğini belirtebiliriz. Böylece **PRINT** deyiminde olduğu gibi "..." içinde verilen mesajlar olduğu gibi görüntülenir. Programı bu kez, bu şekilde yapalım:

```
10 INPUT "KENARI GIRINIZ";A
20 INPUT "YÜKSEKLİĞİ GIRINIZ";H
30 LET ALAN=A*H/2
40 PRINT "ÜÇGENİN ALANI";ALAN;"METREKAREDİR."
50 END
```

Programı çalıştıralım:

```
RUN ↵
KENARI GIRINIZ ? 40 ↵
YÜKSEKLİĞİ GIRINIZ ? 20 ↵
ÜÇGENİN ALANI 400 METREKAREDİR.
Ok
```

Gördüğümüz gibi, programı çalıştırdığımızda bize ne gireceğini sordu. Biz de soruya gerekeni girdik ve sonucu da anlamlı bir biçimde aldık.

Açıklayacak olursak; programı çalıştırdıktan sonra karşımıza **KENARI GİRİNİZ ?** mesajı, biz koymadığımız halde ? işareti ile birlikte çıktı. Biz de değer olarak **40** sayısını girdik ve **ENTER**'e bastık. **YÜKSEKLİĞİ GİRİNİZ ?** sorusunun karşısına da **20** sayısını girdik. **ÜÇGENİN ALANI 400 METREKAREDİR.** çıktısını aldık.

Bu kez de Faiz Problemini yeni gördüğümüz komutlarla ve yanyana girişle çözelim:

```
10 INPUT K,N,T
20 LET F=(K*N*T)/1200
30 PRINT F
40 END
```

Programı çalıştırdığımızda, ekranda ? işareti belirir ve gerekli değerler, aralarına (,) virgül konularak ardı ardına girilir.

```
RUN ↵
? 25000,4,50 ↵
4166.66
Ok
```

Bu programı daha açık şekilde yapalım:

Not: Bundan sonraki programlarımızı **LET** komutunu yazmadan yapacağız.

```
10 INPUT "KAPITALI GIRINIZ";K
20 INPUT "SÜREYİ GIRINIZ (AY)";N
30 INPUT "FAİZ FİYATINI GIRINIZ";T
40 F=(K*N*T)/1200
50 PRINT "PARANIZIN GETİRECEĞİ FAİZ";F;"LİRADIR."
60 END
```

Programı çalıştıralım:

```
RUN ↵
KAPITALI GIRINIZ ? 50000 ↵
SÜREYİ GIRINIZ (AY) ? 6 ↵
FAİZ FİYATINI GIRINIZ ? 50 ↵
PARANIZIN GETİRECEĞİ FAİZ 12500 LİRADIR.
```

Programı çalıştırdığımızda, **KAPİTALİ GİRİNİZ ?** sorusuyla karşılaştık. **50000** değerini girdik, **ENTER**'e bastık. Bu kez **SÜREYİ GİRİNİZ (AY) ?** sorusuna **6** değerini girdik, **ENTER**'e bastık. **FAİZ FİYATINI GİRİNİZ ?** sorusuna da **50** değerini girdik, **ENTER**'e bastık. **PARANIZIN GETİRECEĞİ FAİZ 12500 LİRADIR.** sonucunu aldık.

Bu kez de, basit bir ücret hesaplaması yapan programı yapalım:

```
10 CLS
20 INPUT "ADI SOYADI";AS$
30 INPUT "GÜNLÜK ÜCRET";GU
40 INPUT "ÇALIŞTIĞI GÜN";CG
50 BU=CG*GU           'Brüt Ücret Hesaplanıyor
60 SP=BU*.14         'Sigorta Primi Hesaplanıyor
70 OIND=CG*3000     'Özel İndirim Hesaplanıyor
80 DV=BU*.004       'Damga Vergisi Hesaplanıyor
90 GVM=BU-(SP+OIND) 'Gelir Vergisi Matrahı Hesaplanıyor
100 GV=GVM*.25      'Gelir Vergisi Hesaplanıyor
110 VER=DV+GV       'Vergiler Toplamı Alınıyor
120 NU=BU-(SP+VER)  'Net Ücret Hesaplanıyor
130 PRINT "ADI SOYADI", "BRÜT ÜCRET", "S.S.K.P.", "VERGİLER", "NET ÜCRET"
140 PRINT "=====", "=====", "=====", "=====", "====="
160 END
```

Programı çalıştıralım ve ADI SOYADI'na FİKRET TANIŞ, GÜNLÜK ÜCRET'ine 150000, ÇALIŞTIĞI GÜN'e 24 girelim. Sonuç aşağıdaki gibi olacaktır.

```
RUN ↵
ADI SOYADI      BRÜT ÜCRET      S.S.K.P.      VERGİLER      NET ÜCRET
=====
FIKRET TANIŞ    3600000         504000        770400        2325600
Ok
```

Bu programı her çalıştırdığımızda, istediğimiz kadar işinin ücretini hesaplayabiliriz.

```
10 REM BİLEŞİK FAİZ
20 INPUT "MEVDUAT MIKTARI";K#
30 INPUT "FAİZ";F
40 INPUT "SÜRE";N
50 B#=K#*(1+F/100)^N
60 PRINT :PRINT :PRINT
70 PRINT "SÜRE SONU MEBLAĞ= ";B#
80 END
```

4. PROGRAM İÇERİSİNDEN VERİLERİ İŞLEME READ-DATA

Yazılışı:

```
READ <Değişkenler>,...
"
"
"
DATA <Değer ya da İfadeler>,...
```

Program içerisinden verileri işlemek ve programda kullanılan değişkenlere değer ve ifadeleri atamak için kullanılır. **READ** değişkeni ile aynı **LET** deyiminde olduğu gibi **DATA** satırındaki değer ve ifadeleri okutarak program içerisinden verileri işleyebiliriz.

Bu deyim, program içerisinden verileri işleme açısından **LET** deyimine göre olumlu yanları daha çoktur. Bu nedenle, verileri program içerisinden işleme sırasında daha çok tercih edilir ve kullanılırlar.

READ değişkenleri, **DATA** değerlerini okur ve o **READ** değişkeni **DATA**'daki değeri alır.

```
10 READ A      'DATA deyimindeki 809 sayısı A'ya aktarıldı.
20 DATA 809
30 PRINT A
40 END
```

Bu programda **READ** değişkeni, **DATA**'daki değeri okuyarak **A**'nın değeri **809** oldu.

```
10 READ A,B,C,KŞ,MŞ
...
...
...
80 DATA 5,72,685,"GÖMLEK","ÇORAP"
```

Burada ilk **READ** değişkeni ilk **DATA** değerini alarak işleme koyacaktır.

Bu programı **LET** deyimi ile yapsaydık şöyle olacaktı:

```
10 LET A=5
20 LET B=72
30 LET C=685
40 LET KŞ="GÖMLEK"
50 LET MŞ="ÇORAP"
```

Şimdi, iki programı inceleyecek olursak, birinci programda **READ** değişkenleri ve **DATA** değerleri tek satırda yanyana atanabildi. İkinci programda ise **LET** deyimiyle bu işi daha da uzatarak yaptık.

READ-DATA DEYİMLERİNİN ÖZELLİKLERİ

1. Program içerisinde **READ** ve **DATA** deyimlerinin satır numaralarının önemi yoktur. İlk **DATA** değeri, ilk **READ** değişkenine atanır.

```
10 DATA 5,72
20 READ A
30 READ B,C
40 READ KŞ,MŞ
50 PRINT A,B,C,KŞ,MŞ
60 DATA 685
70 END
80 DATA "GÖMLEK","ÇORAP"
```

Burada **A=5**, **B=72**, **C=685**, **KŞ="GÖMLEK"**, **MŞ="ÇORAP"** değer ve ifadelerini alacaktır.

2. **DATA** deyimindeki veri sayısı **READ** deyimindeki değişken sayısı kadar olmalıdır. **DATA** değerleri fazla olursa, bu fazlalık gözönüne alınmaz. **DATA** değerleri eksik olursa program çalışmaz ve hata mesajı verir.

10 READ A,B	10 READ A,B,C
...	...
...	...
120 DATA 25,46,57	120 DATA 25,46
57 sayısı okunmaz	Hata mesajı verir

Soldaki örneğimizde **DATA** satırındaki üçüncü sayı olan **57** sayısı dikkate alınmayacaktır. Program çalışır, hata mesajı da vermez; ama böyle durumlarda istediğimiz sonucu alamayabiliriz.

Sağdaki örneğimizde ise **READ** satırındaki **C** değişkeni, **DATA** satırından değer alamayarak çalışmayacak ve hata mesajı verecektir.

3. **READ**'in değişken türüne göre **DATA** verisi aynı olmalıdır. **READ** değişkeni sayısal ise **DATA** verisi de sayısal olmalı, **READ** değişkeni alfasayısal ise **DATA** verisi de alfasayısal olmalıdır. **READ** değişkenleri ile **DATA** verileri, arasına virgül konularak ayrılır.

```
10 READ A,B,ADŞ,SOYŞ
...
...
90 DATA 5,66,"RIDVAN","TANIŞ"
```

READ satırındaki değişkenlerle, **DATA** satırındaki veriler birbirine uyuyor. **A** ve **B** değişkenleri sayısal, buna karşılık **5** ve **66** verisi de sayıdır. **ADŞ** ve **SOYŞ** değişkenleri alfasayısal **RIDVAN** ve **TANIŞ** verileri de alfasayısalıdır. Sayısal bir değişkene alfasayısal bir veri, alfasayısal bir değişkene de sayısal bir veri verilseydi program yanlış olurdu.

4. **DATA** komutunun bulunduğu satıra başka bir komut yazılmaz.

```
80 DATA "ISMAIL TANIŞ","ÖĞRETMEN",234,45:PRINT ADŞ
```

Yazılması yanlıştır. Herhangi bir komutun yazılması gerekiyorsa ayrı bir satıra yazılmalıdır.

RESTORE

Programda, **DATA**'daki veriler birden fazla değişkene atanmak istendiğinde **RESTORE** deyimi kullanılır. **RESTORE** deyimi tarafından **DATA**'daki veriler yeniden okutulur.

Yazılışı:
RESTORE <Satır No>

Bu şekilde yazılımda **RESTORE** deyiminden sonra verilen satırdaki **DATA** deyimi esas alınarak **READ** değişkenine aktarılır.

İkinci bir kullanım biçimi de tekrar okutulmak istenilen **READ** değişkeninin bir üst satırına **RESTORE** deyimi yalnız başına yazılarak, **DATA** verileri yeni değişkenler için tekrar okutulur.

```
10 CLS
20 READ A,B,ADŞ
30 RESTORE
40 READ K,M,ASŞ
50 PRINT A,B,ADŞ
60 PRINT K,M,ASŞ
70 DATA 55,120,"ÖZLEM"
80 END
```

Burada **READ** deyimiyle A'ya 55, B'ye 120, ADŞ'a ÖZLEM sözcükleri atanmıştır. 30. satırdaki **RESTORE** deyimiyle 40. satırdaki **READ** değişkenlerine de aynı değerler atanarak K=55, M=120, ve ASŞ=ÖZLEM olmuştur.

```
10 CLS
20 READ A,B
30 RESTORE 70
40 READ K,L
50 DATA 10,20
60 DATA 30,40
70 DATA 50,60
80 PRINT A,B
90 PRINT K,L
100 END
```

Burada ilk olarak A'ya 10, B'ye 20 değerleri atanmıştır. 30. satırdaki **RESTORE 70** yazılımla programın akışı 70 nolu **DATA** satırına gönderilmiş ve K'ya 50, L'ye 60 değerleri atanmıştır. 30. satırdaki **RESTORE 70** yazılımdan 70'i kaldırırsak K'ya 10, L'ye 20 değerlerini yeniden atanmış oluruz.

2. SATIR ve SÜTUN BELİRTEREK YAZDIRMA LOCATE

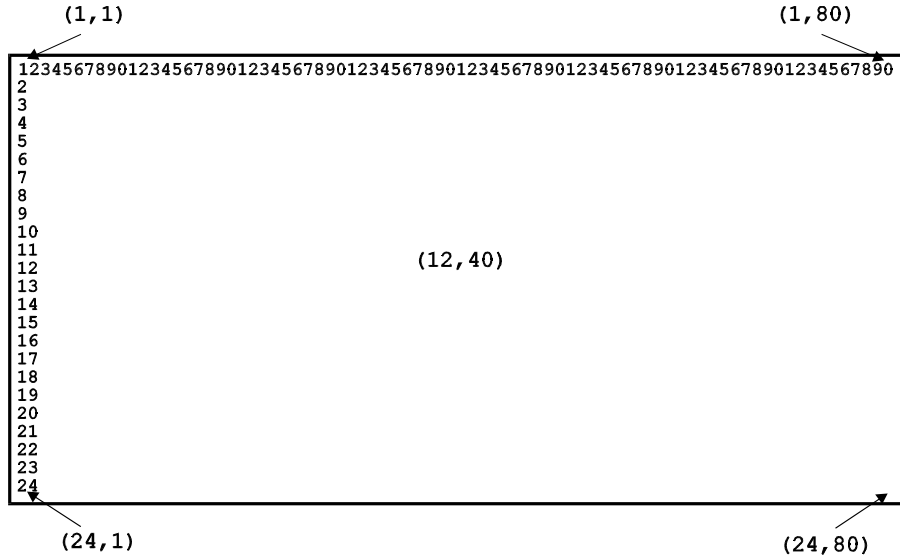
Bu deyim yazılması istenen değer ve sözcükleri, satır ve sütun belirterek ekranın istenilen yerine yazdırmaya yarar.

Yazılışı:
LOCATE <Satır no>,<Sütun no>

LOCATE 10,20 demek ekranın **10.** satırı **20.** sütunu demektir.

Ekranı bir koordinat sistemine benzeterek düşünürsek komutu ve işlevini daha iyi anlayabiliriz.

Aşağıda bir şema ile açıklayacak olursak:



Yukarıda da gördüğümüz gibi ekranın sol üst köşesi (1,1) ile tanımlanıyor. Sağ üst köşe (1,80), orta noktası (12,40), sol alt köşe (24,1), sağ alt köşe (24,80)'dir. Bunları gözönünde bulundurarak basit bir program oluşturalım:

PRINT komutunda ekranın ortasına **MUSTAFA KEMAL ATATÜRK** yazdırmayı görmüştük. Yeni öğrendiğimiz komutla bunu daha kısa yoldan yapalım:

```
10 CLS
20 LOCATE 12,30
30 PRINT "MUSTAFA KEMAL ATATÜRK"
40 END
```

Bu programı çalıştırdığımızda ekranın ortasına **MUSTAFA KEMAL ATATÜRK** yazacaktır.

Ekranı tanımlamaya yarayan bir örnek daha yapalım:

```
10 CLS:KEY OFF ' Alt kısımdaki düğmeler kapatılıyor.
20 LOCATE 1,1:PRINT "SOL ÜST KÖŞE"
30 LOCATE 1,65:PRINT "SAĞ ÜST KÖŞE"
40 LOCATE 12,35:PRINT "ORTA NOKTA"
50 LOCATE 23,1:PRINT "SOL ALT KÖŞE"
60 LOCATE 23,65:PRINT "SOL ÜST KÖŞE"
70 LOCATE 24,30:INPUT "GÖRÜNÜZ VE BİR TUŞA BASINIZ";YŞ
80 IF YŞ="0" THEN 90
90 END
Ok
```

Not: 10. satırda **KEY OFF** komutu ile fonksiyon tuşlarını gösteren anahtarların kapatılmasını sağladık.

3. SAYISAL DEĞERLERİ İSTENEN KALIPTA YAZDIRMA PRINT USING

Yazılışı:

```
PRINT USING <"Alan Tanımlayıcı">;<Değişkenler>
```

PRINT USING deyimiyile yazılması istenen değer ve ifadeler istenilen kalıpta yazdırılabilir.

Dizgisel (String) Karakterler İçin Kullanılan Alan Tanımlayıcıları:

- !** Yalnızca ilk karakteri gösterir.
- \n** **n** adet boşluk \ Alan tanımlayıcısından sonra \ işaretlerinin arasında bırakılan boşluk sayısının **2** fazlası kadar karakteri görüntüler. (**2+n Karakter**)
- &** Alanın uzunluğu kadar karakter görüntüler.

```
10 X$="BILGI":Z$="SAYAR"
20 PRINT USING "!" ;X$;Z$
30 PRINT USING "\  \" ;X$;Z$
40 PRINT USING "\  \" ;X$;Z$;"!!"
50 PRINT USING "!" ;X$;
60 PRINT USING "&" ;Z$
```

```
RUN ↵
BS
BILGISAYAR
BILGI SAYAR !!
BSAYAR
```

Sayısal Değerler İçin Kullanılan Alan Tanımlayıcıları:

- # Sayı işareti olarak tanımladığımız bu işaret, yazdırılacak sayının konumunu tanımlar. Bu işaret kullanıldığında, belirtilen alan doldurulur. Eğer verilen sayı alandan az ise sayı, sağa dayalı olarak yazdırılır.

```
PRINT USING "##.##" ;.45;3.45;5.43
0.45 3.45 5.43
```

```
PRINT USING "###.##" ;457.344;457.346
457.34457.35
```

```
PRINT USING "##.##" ;16.4;8.2;75.678;.234
16.40 8.2075.68 0.23
```

- + Alan tanımlayıcısının önüne ve arkasına konulabilir. Böylece bu işaretin sayının önüne ya da arkasına yazdırılması sağlanır.
- Bu işaret yalnızca - (eksi) değeri olan sayıların sonuna yazılmasını sağlar.

```
PRINT USING "+##.## " ; -74.67;7.3;86.7;-.8
-74.67 +7.30 +86.70 -0.80
```

```
PRINT USING "##.##- " ; -73.88;25.778;-8.03
73.88- 25.78 8.03-
```

- ** Alan tanımlayıcısının önünde yer alır. Yazılan sayısal değer önünde varsa, kalan boşluk kadar * konulmasını sağlar.
- \$\$ İki adet \$ işareti, sayıların önüne \$ işareti konulmasını sağlar.
- **\$ Bu işaretlerle belirlenen alan tanımlayıcısının önüne, sayılardan kalan kalan boşluk kadar * konularak, \$ eklenmesini sağlar.

```
PRINT USING "***#. # ";26.68;- .6;875.3
**26.68 ** -0.60 *875.30
```

```
PRINT USING "$$###.##";650.59
$650.59
```

```
PRINT USING "***$###.##";4.28
***$4.28
```

^^^^ Sayının üslü yazılmasını sağlar. (Yani üslü görüntülenmesini sağlar.) Alan tanımlayıcıdan sonra kullanılır. **E+nn** veya **D+nn** şeklinde yazılacağını gösterir.

, Virgül işareti konulursa rakamları üçer üçer (,) virgülle ayırır.

. Nokta ondalık sayıyı ayırmak için kullanılır. Bir ifadede birden çok kullanılmaz.

```
PRINT USING "####.##,";2583.5
2583.50,
```

```
PRINT USING "####, .##";3896.5
3,896.50
```

```
PRINT USING "##.##^^^^";478.87
4.79E+02
```

```
PRINT USING ".#####^^^^";-747777
%- .7478E+06
```

```
PRINT USING "+.##^^^^";374
+.37E+03
```

```
PRINT USING "_!##.##_!";37.64
!37.64!
```

```
PRINT USING "##.##";453.43
%453.43
```

```
PRINT USING ".##";.999
%1.00
```

% İşareti alan tanımlayıcısından fazla bir sayı verildiğinde çıkar.

D. SİSTEM KOMUTLARI

1. SATIRLARDA DÜZELTME YAPMA

EDIT

Düzeltilme anlamındadır. Program içerisinde satırlarda kısmen veya tamamen değişiklik yapılmak istendiğinde kullanılır. **SATIR EDITÖRÜ** ve **EKRAN EDITÖRÜ** olmak üzere ikiye ayrılır. Yanlışları düzeltmek ve ekleme yapmak amacıyla kullanılır. Yanlış olan satır bu komutla çağırılarak gerekli düzeltme yapılır.

```
10 CLS
20 INPUT "KENARI GIRINIZ ;A
30 INPUT "YÜKSEKLİĞİ GIRINIZ" ;H
40 ALAN=A+H/2
50 PRINT ALAN
60 END
```

Üçgenin alanını hesaplayan, bu programı çalıştırdığımızda **20.** satırda hata mesajı vererek yanlışın üzerinde duracaktır. Çünkü **INPUT** deyimi ile yazdığımız mesajın sonunda " (çift tırnak) koymayı unuttuk. Bu düzeltmeyi yapabiliriz. Bu satırdan sonra program çalışmasına devam eder. Ama sonuç yanlış çıkar. Üçgenin alan formülü $ALAN=(A*H)/2$ 'dir. Yani bizim yaptığımız programın **40** nolu satırı yanlıştır. Şimdi gerekli düzeltmeyi yapalım:

```
EDIT 20 ↵
```

```
20 INPUT _"KENARI GIRINIZ ;A
```

Yazarak imleç yanlışın üzerinde duracaktır.

```
EDIT 40 ↵
```

```
40 ALAN=A+H/2
```

Satırı karşımıza gelir. Gerekli edit tuşlarıyla gereken düzeltmeyi yapar, **ENTER** tuşuna basarız.

Herhangi bir düzeltme veya ekleme durumunda, satırı belleğe göndermek için mutlaka **ENTER** tuşuna basmalıyız.

EDIT komutuyla düzeltme yapmamızın bir yararı da, yaptığımız yanlış **BASIC** kurallarına aykırı ise imleç, yanlışın üzerinde durarak bize yol gösterecektir.

Yukarıda yaptığımız işleme satır editi denir. Bilgisayarlarda ekran editörü de vardır. Ekran editinde liste ekranda iken, edit tuşları ile imleci istenen satıra götürerek gereken düzeltme yapılır. Düzeltme yapıldıktan sonra **ENTER** tuşuna basılarak satırdan çıkılır.

2. SATIR NUMARALARINI YENİDEN DÜZENLEME RENUM

Bellekteki programın satır numaralarını yeniden düzenlemek için kullanılır. **RENUMBER**'in kısa yazılışıdır.

Yazılışı:
RENUM [İlk satır no], [Uygulamaya başlanacak satır no],
[Satır artış deĞeri]

Örnek: RENUM 100,5,10

<u>ESKİ SATIR NO</u>	<u>YENİ SATIR NO</u>
5	100
10	110
15	120
25	130
30	140
37	150

Eski programın 5. satırının 100 olmasını ve 10'ar artmasını istedik. Programın yeni satır numaraları 100'den başlayarak 10'ar arttı.

Bellekte şu satırların yüklü olduğu bir programı istediğimiz şekle çevirme örneklerini aşağıda görelim:

5,10,12,20,25,30,35,38,40,50,60,65,66,67,80,90

RENUM ↵

Programın satır nolarını 10'dan başlatır, 10'ar artırır.

10,20,30,40,50,60,70,80,90,100,110,120,130,140,150,160 şekline dönüşür.

RENUM 50,,20 ↵ 50.'den başlar 20'şer arttırır.

50,70,90,110,130,150,170,190,210,230,250,270,290,310,330,350

RENUM 100,50,10 ↵ **50.** satır **100** olmak üzere **10'**ar artar.

5,10,12,20,25,30,35,38,40,100,110,120,130,140,150,160,170

3. OTOMATİK SATIR NUMARASI VERME

AUTO

Program yazımı sırasında otomatik olarak satır numarası üretir. Program yazmaya başlarken direkt mod'da yazılır.

Yazılışı:

AUTO ↵ Satır numaraları **10'**dan başlar ve **10'**ar artar.
AUTO X,Y ↵ **X** ile belirtilen satır no'dan başlayıp, **Y** ile belirtilen değerle artar.
AUTO 100,20 ↵ **100.** satırdan başlar, **20'**şer artarak gider.

Otomatik yazımdan çıkmak için **Ctrl+C** tuşlarına basmamız gerekir.

4. SESLİ UYARMA

BEEP

Program içerisinde ses üretmek için kullanılır.

```
10 CLS
20 PRINT "FIKRET TANIŞ"
30 BEEP
40 PRINT "KAMURAN TANIŞ"
50 BEEP
60 END
```

5. MS-DOS'A GEÇİCİ OLARAK ÇIKMA

SHELL

Yazılışı:

SHELL

Basic ortamından sisteme dönerek, **DOS** komutlarını kullanmayı sağlar. Basic'te çalışırken **SYSTEM** komutuyla çıkma yerine **SHELL** ile kısa bir süre **DOS'a** dönülerek işlemlerimizi yürütürüz. Basic'e geri dönüş için **EXIT** deyimi kullanılır.

Program içerisinde **SHELL** komutu kullanılmak isteniyorsa, satır numarasından sonra **SHELL "MS-DOS Komutu"** şeklinde yazıldığında, **DOS'a** geçer ve verilen görevi yerine getirdikten sonra kendiliğinden **BASIC** ortamına geçer.

40 SHELL "DIR" yazdığımızda **40.** satırda **DOS'a** geçerek, dosya listelerini verir ve Basic ortamına döner.

6. MS-DOS'DAN BASIC'E GERİ DÖNÜŞ

EXIT

Yazılışı:
EXIT

SHELL deyimiyile basic ortamından çıkıldıktan sonra **DOS'tan** tekrar, basic ortamına döndürür.

Örneğin basic ortamında **SHELL** yazıp Enter'e basalım:

```
SHELL ↵
```

A> promptu çıkar. Yanına **DIR** yazarak işletim sisteminde dosya rehberini listeyelim. İşletim sisteminde işlemiz bittikten sonra **EXIT** yazıp Basic'e dönelim.

```
A>EXIT ↵  
Ok
```

İle tekrar basic ortamına döneriz.

E. KONTROL KOMUTLARI

1. KOŞULSUZ GÖNDERME

GOTO

Yazılışı:
GOTO <Satır Numarası>

Örnek:
GOTO 140

Bir **BASIC** programında, programın işleyişini başka bir satıra göndermek gerekir. **GOTO** komutu programı çalışması sırasında şartsız olarak başka bir satıra gönderir. **GOTO** deyiminden sonra programın devam edeceği satır numarası yazılır. **GOTO 70** gibi. Bu şekilde yazılım programın akışını **70.** satıra gönderir.


```
10 PRINT "BIRINCI SATIR"  
20 PRINT "IKINCI SATIR"  
30 PRINT "ÜÇÜNCÜ SATIR"  
40 GOTO 70  
50 PRINT "BEŞİNCİ SATIR"  
60 PRINT "ALTINCI SATIR"  
70 PRINT "YEDİNCİ SATIR"  
80 PRINT "SEKİZİNCİ SATIR"  
90 PRINT "DOKUZUNCU SATIR"  
100 PRINT "ONUNCU SATIR"  
110 END
```

Programı çalıştırdığımızda aşağıdaki sonucu görürüz:

```
RUN ↵  
BIRINCI SATIR  
IKINCI SATIR  
ÜÇÜNCÜ SATIR  
YEDİNCİ SATIR  
SEKİZİNCİ SATIR  
DOKUZUNCU SATIR  
ONUNCU SATIR  
Ok
```

Program ilk üç satırda istenen yazma görevini yerine getirerek **40.** satıra geldiğinde **GOTO 70** deyiimiyle **50.** ve **60.** satırları atlayarak **70.** satıra gider; çalışmasına **70.** satırdan itibaren devam etmiştir.

```
10 PRINT "PAZARTESİ"  
20 PRINT "SALI"  
30 PRINT "ÇARŞAMBA"  
40 PRINT "PERSEMBE"  
50 PRINT "CUMA"  
60 PRINT "CUMARTESİ"  
70 PRINT "PAZAR"  
80 GOTO 130  
90 PRINT "ILKBAHAR"  
100 PRINT "YAZ"  
110 PRINT "SONBAHAR"  
120 PRINT "KIŞ"  
130 END
```

Yukarıdaki program çalışırken, normal olarak **70.** satıra kadar haftanın günlerini yazacak; **80.** satıra geldiğinde **GOTO 130** deyiimiyle **130.** satıra gidecek **130.** satırda **END** deyiimi bulunduğundan, mevsimleri yazmadan program sona erecektir.

```
10 PRINT "ÖRNEK"  
20 CLS  
30 GOTO 10
```

Bu programı çalıştırdığımızda **10.** satırda **ÖRNEK** yazacak, **20.** satırda ekranı temizleyecek, **30.** satıra geldiğinde tekrar **10.** satıra giderek program sürecektir. Bu olay o kadar hızlı olacaktır ki **ÖRNEK** sözcüğü ekranda yanıp söniyor gibi görünecektir. Bu program sonsuz döngüye girdiği için **CONTROL+C** veya **CONTROL+BREAK** ile çıkmamız gerekecektir.

Aynı programda **20.** satırı silerseniz, **ÖRNEK** sözcüğünü alt alta sonsuza kadar yazacaktır. Programın sona ermesi için, yine dışarıdan müdahale edilmesi gerekecektir.

READ-DATA deyimleri ile herhangi **10** sayıyı okutan ve ekrana yazdıran programı yapalım:

```
10 CLS  
20 READ A  
30 IF A=-1 THEN 60  
40 PRINT A  
50 GOTO 20  
60 END  
70 DATA 8,6,12,4,20,4,16,9,3,19,-1
```

Burada **30.** satır kontrol satırıdır. **A** değişkeni **-1** değerini aldığı zaman program sona erecektir. Dikkat ettiyseniz **DATA** satırında son olarak **-1** verisini verdik. **-1** bizim için burada kullanılmayacak bir sayı olarak düşünüldü.

Aynı programı sayıların karelerini alacak şekilde yeniden yapalım:

```
10 CLS  
20 READ A  
30 IF A=-1 THEN END  
40 KARE=A^2  
50 PRINT A,KARE  
60 GOTO 20  
70 DATA 8,6,12,4,20,4,16,9,3,19,-1
```

Bu kez de **READ-DATA** ile 20 kişilik bir sınıfta kaçınıcı öğrencinin notu isteniyorsa o öğrencinin notunu bulan ve yazan programı yapalım:

```
10 CLS
20 S=0
30 INPUT "KAÇINCI ÖĞRENCİNİN NOTU";K
40 S=S+1
50 READ N
60 IF K=S THEN PRINT K;"INCI ÖĞRENCİNİN NOTU";N;"DIR":END
70 GOTO 40
80 DATA 5,6,7,7,9,3,5,2,8,6,5,5,10,7,8,9,3,4,10,7
```

2. KOŞULLU SAPMA

IF-THEN-ELSE

Programın akışını koşullu olarak değiştirmeye yarar. Bu deyimle istenen koşulun yerine getirilmesi durumunda, koşul doğrultusunda programın akışı sürer. Koşul yerine getirilmediği takdirde ya istediğimiz sonucu alamayız ya da bir alt satıra geçerek programın akışı sürer.

Bu deyimlerden sonra şu karşılaştırma operatörlerini kullanarak istenilen işlemleri yapması sağlanır. (=,>,<,<=,>=,<>)

Yazılışı:

```
IF <Değişken>=<Bir İfade> THEN <Basic Deyimi> ELSE
<Basic
```

Deyimi>

```
IF A=0 THEN GOTO 120
Eğer A=0 İse Git 120
ya da
IF A=100 THEN PRINT "MERHABA" ELSE END
Eğer A=100 İse Yaz MERHABA Değilse Bitir
```

Aşağıdaki örnekte bu komutlarla yapacağımız programla işleyişini görelim:

```
5 CLS
10 INPUT "BİR HARF GIRINIZ ";HŞ
20 IF HŞ="M" THEN GOTO 50
30 IF HŞ="G" THEN GOTO 70
40 PRINT "YANLIŞ GİRİŞ YAPTINIZ":END
50 PRINT "MERHABA"
60 GOTO 10
70 PRINT "GÜNAYDIN"
80 GOTO 10
```

Bu programa göre kullanıcıdan bir harf girmesi istenmektedir. Girilen harf **M** ise **20.** satır tarafından koşul sağlandığı için **50.** satıra gönderilecektir. **50.** satırda **MERHABA** sözcüğünü yazmasını istediğimizden, **MERHABA** yazarak **60.** satır tarafından tekrar **10.** satıra gönderilerek, yeniden bir harf girmesi istenecektir. Girilen harfin **G** olması durumunda **20.** satırda koşul yerine getirilmediği için, bir alt satıra yani **30.** satıra geçerek orada istenen koşulu dikkate alacaktır. **30.** satırda **HŞ**'in ifadesi **G** olduğundan koşul yerine getirilmiş olacak ve satırın gereği olarak **70.** satıra gidecektir. **70.** satırda **GÜNAYDIN** yazması istendiğinden, gereğini yerine getirerek **80.** satır kanalıyla tekrar başa dönecektir.

Girilen harf **M** veya **G** dışında bir harfse, **20.** ve **30.** satırlarda koşul sağlanmadığından bir alt satıra, yani **40.** satıra geçecek **YANLIŞ GİRİŞ YAPTINIZ** yazarak program sona erecektir.

Buradan şunu anlıyoruz: Koşul sürdürdüğümüz satırda, istenilenler sağlanıyorsa gereken yapılacaktır. Eğer koşul sağlanmıyorsa, bir alt satırı okuyacaktır.

Aynı programı şu şekilde de yapmak mümkündür:

```
10 INPUT "BİR HARF GIRINIZ";HŞ
20 IF HŞ="M" THEN PRINT "MERHABA":GOTO 10
30 IF HŞ="G" THEN PRINT "GÜNAYDIN":GOTO 10
40 PRINT "YANLIŞ GİRİŞ YAPTINIZ":END
```

Bu kez de aynı örneğin benzerini sayısal değerlerle yapalım: Bu programımız da **1**'den **5**'e kadar olan sayıları yazı ile yazsın.

```
10 CLS
20 INPUT "1 İLE 5 ARASINDA BİR SAYI GIRINIZ";S
30 IF S=1 THEN PRINT "BİR"
40 IF S=2 THEN PRINT "İKİ"
50 IF S=3 THEN PRINT "ÜÇ"
60 IF S=4 THEN PRINT "DÖRT"
70 IF S=5 THEN PRINT "BEŞ"
80 IF S>5 THEN PRINT "YANLIŞ GİRİŞ YAPTINIZ":GOTO 20
90 END
```

Bu programın çalışmasını inceleyelim ve ikisinin arasındaki farka bakalım: İlk programımızda girilen bilgiler alfasayısal olduğu için **HŞ="G"** gibi koşulları **"..."** içinde yazdık. İkincisinde ise sayısal olduğu için **".."** içinde yazmadık. Yani **S=1** yazdık. Bundan sonraki programlarımızda bu noktaları unutmamalım.

```
10 ' DÖRT İŞLEM YAPAN PROGRAM
20 INPUT "BİRİNCİ SAYIYI GIRINIZ";A
25 IF A=0 THEN END
30 INPUT "İŞLEMI İŞARET OLARAK GIRINIZ";CŞ
40 INPUT "İKİNCİ SAYIYI GIRINIZ";B
50 IF CŞ="+" THEN 90
60 IF CŞ="-" THEN 120
70 IF CŞ="/" THEN 150
80 IF CŞ="*" THEN 180
90 D=A+B
100 PRINT "TOPLAMANIN SONUCU";D
110 GOTO 10
120 D=A-B
130 PRINT "ÇIKARMANIN SONUCU";D
140 GOTO 10
150 D=A/B
160 PRINT "BÖLMENİN SONUCU";D
170 GOTO 10
180 D=A*B
190 PRINT "ÇARPMANIN SONUCU";D
200 GOTO 10
```

3. BASIC DİLİNDE BAĞLAÇLAR

AND-OR-NOT

(VE-VEYA-DEĞİL)

Koşullu sapma komutlarında, **AND-OR-NOT** bağlaçlarını kullanarak koşullarımızı daha net şekilde vermeyi sağlayabiliriz.

Kullanımları **IF-THEN-ELSE** komutlarıyla birliktedir. Bu komutlarla birlikte kullanarak birden çok olasılıkları da gözönüne alabiliriz.

Aşağıdaki tabloda, **AND-OR-NOT** deyimlerinin çalışma şemasını bulacaksınız. **D** doğruyu, **Y** yanlış göstermektedir.

NOT	
X	NOT X
D	Y
Y	D

AND		
X	Y	X AND Y
D	D	D
D	Y	Y
Y	D	Y
Y	Y	Y

AND deyiminde her iki ifadenin de doğru olduğu durumda, X ve Y ikisinde doğru ise sonuç doğru olur. Aksi durumda yanlış olur.

OR		
X	Y	X OR Y
D	D	D
D	Y	D
Y	D	D
Y	Y	Y

OR deyiminde her iki ifadenin de yanlış olduğu durumda, X ve Y ikisi de yanlış ise sonuç yanlıştır. Aksi durumda doğrudur.

IF A>1 AND A<20 THEN GOTO 100

Eğer A büyük 1 ve A küçük 20 ise 100. satıra git

IF A>1 OR B<=20 THEN GOTO 120

Eğer A büyük 1 veya B küçük eşit 20 ise 120. satıra git

IF NOT A=0 THEN 140

Eğer A, Sıfıra eşit değilse 140. satıra git

IF A=>65 AND A=<74 THEN PRINT "IYI"

Eğer A Eşit 10 ise PEKİYİ yaz

IF A>10 OR B>20 THEN C=1

Eğer A büyük 10 ya da B>20 ise C, 1'e eşit olsun

IF Y\$="E" OR Y\$="e" THEN 10 ELSE END

Eğer Y\$, E ya da Y\$ e, ise 10 satıra git. Değilse programı sona erdir.

IF Y\$="B" AND S=0 THEN INPUT "ADINIZI GIRINIZ";AD\$

Eğer Y\$, B ve S, 0 ise ADINIZI GİRİNİZ girişini yap

IF A<>0 AND B<>100 THEN 200

Eğer A sıfıra eşit değilse ve B 100'e eşit değilse 200. satıra git

IF (A<=6 AND C\$="K") OR (A<=7 AND C\$="E") THEN 300

Eğer A küçük eşit 6 ve C\$=K veya A küçük eşit 7 ve C\$=E ise 300. satıra git.

Yukarıda verdiğimiz örnekleri, yapacağımız programın akışına göre istediğimiz kadar çoğaltabilir ve uyarlayabiliriz.

Örnek: Bir gençlik kulübüne girebilmek için kızların yaşının 18'den, erkeklerin yaşının 17'den büyük olması ve kulübün parolasını bilmesi gerekmektedir. Bu koşullar sağlanıyorsa, İÇERİ GİREBİLİRSİNİZ; sağlanmıyorsa İÇERİ GİREMEZSİNİZ yazıp tekrar başa dönen programı yapalım. Parola GENÇ KUŞAK'dır.

```
10 INPUT "YAŞINIZI GIRINIZ:";Y
20 INPUT "CINSİYETİNİZİ GIRINIZ:";CŞ
30 INPUT "PAROLAYI GIRINIZ:";PŞ
40 IF CŞ="K" AND Y>18 AND PŞ="GENÇ KUŞAK" THEN GOTO 80
50 IF CŞ="E" AND Y>17 AND PŞ="GENÇ KUŞAK" THEN GOTO 80
60 PRINT "İÇERİ GİREMEZSİNİZ"
70 GOTO 10
80 PRINT "İÇERİ GİREBİLİRSİNİZ"
90 GOTO 10
```

10, 20 ve 30 numaralı satırlarda kulübe girmek isteyenlerin yaşı, cinsiyeti ve parolayı girmesi istenmektedir. 40. ve 50. satırlarda koşul verilmiş ve verilen koşulların yerine getirilmesi durumunda 80. satıra gönderilerek İÇERİ GİREBİLİRSİNİZ yazıp başa dönecektir. Koşullar yerine getirilmediği takdirde 60. satırda İÇERİ GİREMEZSİNİZ yazarak yine başa dönerek, yeni girişlerin yapılması gerçekleştirilecektir. Bu programın 40. ve 50. satırını birleştirerek tek bir satır haline getirebiliriz.

```
40 IF (CŞ="K" AND Y>18 AND PŞ="GENÇ KUŞAK") OR (CŞ="E" AND
Y>17 AND PŞ="GENÇ KUŞAK") THEN 80
```

40. satırını yukarıdaki biçimde yazıp, 50. satırını da silerseniz aynı sonucu elde ederiz.

Vergi İadesini hesaplayan program.

```
10 CLS
20 INPUT "ADI SOYADI";ADŞ
30 INPUT "MAAŞI";MAAS#
40 INPUT "TOPLADIĞI FATURA TUTARI";TF#
50 IF TF#>MAAS# THEN TF#=MAAS#
60 IF TF#<=60000 THEN 100
70 IF TF#>60000 AND TF#<=120000 THEN 110
80 IF TF#>120000 AND TF#<=200000 THEN 120
90 IF TF#>200000 THEN 130
100 IADE!=TF#*.10: GOTO 140
110 IADE!=6000+(TF#-60000)*.2:GOTO 140
120 IADE!=18000+(TF#-120000)*.12:GOTO 140
```

```
130 IADE!=27600+(TF#-200000)*.05:GOTO 140
140 DV=IADE!*.005
150 NETIAD#=IADE!-DV
160 CLS
170 PRINT "ADI SOYADI", "MAAŞ", "T.FATURA", "D.VERGISI",
"NET IADE"
180 PRINT "=====", "====", "=====", "=====",
"====="
190 PRINT AD$, MAAS#, TF#, DV, :PRINT USING "###,###";NETIAD#
```

4. ALT PROGRAMLAR (SUBROUTINE)

a. GOSUB-RETURN

Yazılışı:

```
GOSUB <Alt Program Satır Numarası>
...
...
Alt program
...
...
RETURN
```

Program içerisinde göndermeleri, şimdiye kadar **GOTO** deyimiyle yaptık. Ama bazı durumlar olur ki, yaptığımız bir alt programı, sık sık kullanmamız gerekebilir. Bunlar dosya açma, ekran çizimi gibi alt programlardır. İşte böyle durumlarda **GOSUB-RETURN** komutlarından yararlanırız. Aynı programı yeniden, (aynı program içinde yapmak yerine), bir kez yaparak defalarca kullanabiliriz.

GOSUB deyimi, **GOTO SUBROUTINE** deyiminin kısa yazılışıdır. **GOSUB** deyiminden sonraki **Satır No**, alt programın satır numarasıdır. Bu nedenle programın akışını alt programa götürür.

RETURN alt programın son deyimidir. Bu deyim programın akışını geldiği **GOSUB** deyiminin bir alt satırına gönderir.


```
10 REM ANA PROGRAM BAŞLANGICI
20 CLS
30 GOSUB 500
40 ...
50 ...
60 ...
70 END
...
...
...
500 REM ALT PROGRAM BAŞLANGICI
510 ...
520 ...
530 ...
540 RETURN
```

Böyle bir program çalışmaya başladığında, **30.** satırdaki **GOSUB 500** deyimiyile **500.** satırda bulunan **Alt Program'a** gider. Program bundan sonra buradan çalışmaya devam eder. **RETURN** deyimini görene kadar bu işlem sürer. **RETURN** deyimini gördükten sonra geldiği yer olan **30.** satırın bir altına döner. Bu da **40.** satırdır.

```
10 CLS
20 GOSUB 70
30 GOSUB 100
40 GOSUB 130
50 GOSUB 160
60 END
70 PRINT "BİRİNCİ ALT PROGRAM"
80 PRINT
90 RETURN
100 PRINT "İKİNCİ ALT PROGRAM"
110 PRINT
120 RETURN
130 PRINT "ÜÇÜNCÜ ALT PROGRAM"
140 PRINT
150 RETURN
160 PRINT "DÖRDÜNCÜ ALT PROGRAM"
170 PRINT
180 RETURN
```

```
RUN ↵
BİRİNCİ ALT PROGRAM

İKİNCİ ALT PROGRAM

ÜÇÜNCÜ ALT PROGRAM

DÖRDÜNCÜ ALT PROGRAM
Ok
```

Gördüğümüz gibi alt programlara geçen program akışı, **RETURN** deyimiyle alt programın bir alt satırına döndü ve tekrar oradan itibaren çalışmaya başladı. Orada da yine alt programa gönderme olduğundan gönderildiği alt programa gitti ve program çalışmasına böyle devam etmiş oldu.

b. ON...GOTO / ON...GOSUB DEYİMLERİ

Yazılışı:

```
ON <Ifade> GOTO <SatırNo 1>,<SatırNo 2>,...<SatırNo N>
ON <Ifade> GOSUB <SatırNo 1>,<SatırNo 2>,...<SatırNo N>
```

Buradaki ifade sayısal ifadedir. Bu ifadenin aldığı değere göre, belirlenen satır no'ya gider. **İfade=1** ise programın akışı **SatırNo 1** ile belirtilen satır numarasına gider, **İfade=2** ise programın akışı **SatırNo 2** ile belirtilen satır no'ya gider.

```
10 PRINT "1- YAZIYLA"
20 PRINT "2- YAZIYLA"
30 PRINT "3- YAZIYLA"
40 PRINT "4- PROGRAMDAN ÇIKIŞ"
50 INPUT "SEÇİMİNİZ";S
60 ON S GOTO 70,80,90,100
70 PRINT "BİR BASTINIZ":GOTO 10
80 PRINT "İKİ BASTINIZ":GOTO 10
90 PRINT "ÜÇ BASTINIZ":GOTO 10
100 PRINT "PROGRAMDAN ÇIKTINIZ":END
```

Bu programı çalıştırdığımızda, karşımıza bir menü çıkacaktır. Bu menüde **4** seçenek vardır. Biz bu seçeneklerden herhangi birine basarsak, programda yer alan **ON S GOTO** deyiminden sonra gelen satır numaraları bu değerlere göre program akışını yönlendirecektir. Yani **1** bastığımızda **80.** satıra, **2** bastığımızda **90.** satıra gidecektir ve yazı ile bize kaç bastığımızı bildirecektir.

Faiz hesabını **GOSUB...RETURN** ile oluşturarak, hesaplamasını istediğimiz alt programa götüren, programı yapalım:

```
10 ' ***** ANA MENÜ EKRAM OLUŞUMU *****
20 CLS:KEY OFF      ' Aşağıdaki FONKSİYON TUŞ'ları kapatılıyor.
30 GOSUB 630      ' Bu satırlar ÇERÇEVE ÇİZİMİ ALT PROGRAMINA gidiyor
40 LOCATE 5,15:COLOR 0,7:PRINT "***** FAİZ HESABI ANA MENÜSÜ
*****":COLOR 7,0
50 LOCATE 8,20:PRINT "1. KAPITAL HESABI"
60 LOCATE 10,20:PRINT "2. VADE HESABI"
70 LOCATE 12,20:PRINT "3. FAİZ FİYATI HESABI"
80 LOCATE 14,20:PRINT "4. FAİZ HESABI"
90 LOCATE 16,20:PRINT "5. PROGRAMDAN ÇIKIŞ"
100 LOCATE 20,25:PRINT "SEÇENEĞİNİZ...>"
110 LOCATE 20,42:INPUT " ",S      ' INPUT komutundan sonra boş tırnak
120 IF S<1 OR S>5 THEN 100      ' açılır ve VİRGÜL konursa SORU
130 ON S GOTO 140,250,360,470,580      ' İŞARETİ (?) görüntülenmez
140 GOSUB 630
150 ' ***** KAPITAL HESABI ALT PROGRAMI *****
160 LOCATE 5,15:COLOR 0,7:PRINT "**** KAPITAL HESABI ****":
COLOR 7,0
170 LOCATE 8,20:INPUT "VADEYİ GIRINIZ";N
180 LOCATE 10,20:INPUT "FAİZ FİYATINI GIRINIZ";T
190 LOCATE 12,20:INPUT "FAİZİ GIRINIZ";F
200 K=(F*1200)/(N*T)
210 LOCATE 16,15:PRINT "BU KOŞULLARDA
KAPİTALİNİZ";K;"LİRADIR"
220 LOCATE 22,25:COLOR 0,7:PRINT "DEVAM İÇİN BİR TUŞA
BASINIZ":COLOR 7,0
230 IF INKEY$="" THEN 230
240 GOTO 10
250 GOSUB 630
260 ' ***** VADE HESABI ALT PROGRAMI *****
270 LOCATE 5,15:COLOR 0,7:PRINT "**** VADE HESABI ****":
COLOR 7,0
280 LOCATE 8,20:INPUT "KAPİTALI GIRINIZ";K
290 LOCATE 10,20:INPUT "FAİZ FİYATINI GIRINIZ";T
300 LOCATE 12,20:INPUT "FAİZİ GIRINIZ";F
310 N=(F*1200)/(K*T)
320 LOCATE 16,15:PRINT "BU KOŞULLARDA PARANIZ";N;"AY FAİZDE
KALMALIDIR"
330 LOCATE 22,25:COLOR 0,7:PRINT "DEVAM İÇİN BİR TUŞA
BASINIZ":COLOR 7,0
340 IF INKEY$="" THEN 340
```

```
350 GOTO 10
360 GOSUB 630
370 ' ***** YÜZDE HESABI ALT PROGRAMI *****
380 LOCATE 5,15:COLOR 0,7:PRINT "***** YÜZDE HESABI *****":
COLOR 7,0
390 LOCATE 8,20:INPUT "KAPITALI GIRINIZ";K
400 LOCATE 10,20:INPUT "VADEYİ GIRINIZ";N
410 LOCATE 12,20:INPUT "FAİZİ GIRINIZ";F
420 T=(F*1200)/(K*N)
430 LOCATE 16,15:PRINT "BU KOŞULLARDA FAİZ FİYATI
%;T;"OLMALIDIR"
440 LOCATE 22,25:COLOR 0,7:PRINT "DEVAM İÇİN BİR TUŞA
BASINIZ":COLOR 7,0
450 IF INKEY$="" THEN 450
460 GOTO 10
470 GOSUB 630
480 ' ***** FAİZ HESABI ALT PROGRAMI *****
490 LOCATE 5,15:COLOR 0,7:PRINT "***** FAİZ HESABI *****":COLOR
7,0
500 LOCATE 8,20:INPUT "KAPITALI GIRINIZ";K
510 LOCATE 10,20:INPUT "VADEYİ GIRINIZ";N
520 LOCATE 12,20:INPUT "FAİZ FİYATINI GIRINIZ";T
530 F=(K*N*T)/1200
540 LOCATE 16,15:PRINT "BU KOŞULLARDA PARANIZ";F;"LİRA FAİZ
GETİRİR"
550 LOCATE 22,25:COLOR 0,7:PRINT "DEVAM İÇİN BİR TUŞA
BASINIZ":COLOR 7,0
560 IF INKEY$="" THEN 560
570 GOTO 10
580 GOSUB 630
590 ' ***** PROGRAMI SONA ERDİRME *****
600 FOR SS=1 TO 50:LOCATE 12,30:PRINT "TEŞEKKÜR EDERİM":
NEXT SS:CLS
610 KEY ON ' FONKSİYON TUŞ'ları yeniden açılıyor.
620 END
630 ' ***** ÇERÇEVE ÇİZİMİ *****
640 CLS
650 LOCATE 3,5:PRINT CHR$(201)
660 LOCATE 3,6:PRINT STRING$(69,205)
670 LOCATE 3,75:PRINT CHR$(187)
680 FOR I=4 TO 21
690 LOCATE I,5:PRINT CHR$(186):LOCATE I,75:PRINT CHR$(186)
700 NEXT I
710 LOCATE 22,5:PRINT CHR$(200)
720 LOCATE 22,6:PRINT STRING$(69,205)
```

```
730 LOCATE 22,75:PRINT CHR$(188)
740 RETURN
```

F. SAYAÇ KULLANIMI

$X=1$ ile başlayan X değerini, bir döngü içerisine sokarak $X=X+1$ ifadesiyle her dönüşünde 1 artırabiliriz. İsteğe göre bu artırma işini $X=X+5$ şeklinde yazarak 5 'er de artırabiliriz. Bu yolla sayı artırma işine Sayaç adını veriyoruz. Bir örnek programla bunu açıklayalım:

1 'den başlayıp 100 'e kadar olan sayıların toplamını alıp, toplamları ekrana yazdıran programı yapalım:

```
10 CLS
20 S=0
30 S=S+1
40 TOPLAM=TOPLAM+S
50 PRINT S,TOPLAM
60 IF S=100 THEN END
70 GOTO 30
```

Bu programı çalıştırdığımızda, 1 'den 100 'e kadar sayıların toplamlarını verecektir. Burada sayaç görevini S değişkeni görüyor. Program çalışmaya başladığında S 'nin değeri 0 'dir. $30.$ satırda S 'ye bir eklenecek ve her geri dönüşte S 'nin değeri 1 artacaktır. $40.$ satırda S değeri her dönüşte **TOPLAM** değişkenine eklenecektir. $50.$ satırda S (Sayı) ve **TOPLAM** ekrana yazılacaktır. $60.$ satır kontrol satırıdır. Burada S , 100 değerini aldığı anda program sona erecektir. $70.$ satır programın akışını $30.$ satıra gönderir. S 'nin değeri her dönüşte artar, **TOPLAM** değişkeninin de değeri buna bağlı olarak artacaktır. Bu işlemler $60.$ satırdaki koşul sağlanıncaya kadar sürecektir. $S=100$ olduğunda program sona erecektir.

Bu kez de ekrana 20 kez **KAMURAN TANIŞ** yazdıran programı yapalım:

```
10 CLS
20 S=0
30 S=S+1
40 PRINT "KAMURAN TANIŞ"
50 IF S=20 THEN END
60 GOTO 30
```

Bu programımızda da sayaç içerisinde **40.** satırda **KAMURAN TANIŞ'** yazmasını istediğimiz için, bunu yerine getirecektir. Bu olay **50.** satırdaki kontrol satırında **S** değişkeninin **20** değerini almasına kadar sürecektir.

0'dan, 5'er artarak 100'e kadar giden sayıların karelerini ve küplerini alan program:

```
10 CLS
20 S=0
30 PRINT "SAYI", "KARESİ", "KÜPÜ"
40 PRINT "====", "=====", "====="
50 S=S+5
60 KARE=S^2      \ya da      60 KARE=S*S
70 KUP=S^3      \ya da      70 KUP=S*S*S
80 PRINT S, KARE, KUP
90 IF S=100 THEN END
100 GOTO 50
```

Bu programımızın üst satırında bir süsleme yaparak, sayıların ne olduğunun görülmesini sağladık.

Yeni bir program yaparak, sayıları azaltmayı sağlayalım. **100'den 0'a** kadar **10'ar** azalarak giden sayıların **KAREKÖK'**lerini alan programı yapalım:

```
10 CLS
20 S=100
30 S=S-10
40 KAREKOK=S^(1/2)
50 IF S=0 THEN END
60 PRINT S, "KAREKÖKÜ", KAREKOK
70 GOTO 30
```

100'den 0'a kadar **10'ar** azalan sayıların kareköklerini alan program:

```
10 ' KLAVYEDEN GIRILEN 10 SAYI İÇİNDEN POZITIF
20 ' SAYILARIN ORTALAMASINI BULAN PROGRAM
30 S=0
40 TOP=0:PS=0
50 INPUT "SAYI=";SAYI
60 S=S+1
70 IF SAYI>0 THEN TOP=TOP+SAYI:PS=PS+1
80 IF S<10 THEN 50
90 ORT=TOP/PS
100 PRINT "POZITIF SAYILARIN ORTALAMASI";ORT
110 END
```

```
10 CLS
20 PRINT "BU PROGRAM "
30 PRINT "İSTENİLEN KADAR ÖĞRENCİ'NİN NOTLARININ GIRILMESİ
İÇİNDİR."
40 PRINT "ÖĞRENCİ SAYISI, ORTALAMA, EN KÜÇÜK VE EN BÜYÜK
NOTU VERİR."
50 PRINT "GİRİŞLERİNİZ BİTTİĞİNDE, ÇIKIŞ İÇİN ÖĞRENCİ NO'YU
<ENTER>'LE BOŞ GEÇİNİZ."
60 PRINT:PRINT :PRINT
70 INPUT "ÖĞR NO:";N
80 IF N=0 THEN 170
90 INPUT "NOT";NN
100 I=I+1
110 TOP=TOP+NN
120 IF I=1 THEN ENB=NN :ENK=NN
130 IF NN>ENB THEN ENB=NN
140 IF NN<ENK THEN ENK=NN
150 ORT=TOP/I
160 GOTO 70
170 CLS
180 PRINT "ÖĞR.SAYISI:";I
190 PRINT "ORTALAMA..:";ORT
200 PRINT "EN KÜÇÜK..:";ENK
210 PRINT "EN BÜYÜK..:";ENB
220 END
```

```
10 ' 10 ÖĞRENCİNİN YAŞI KLAVYEDEN GIRILEREK
20 ' BU ÖĞRENCİLERİN YAŞ ORTALAMASINI VE
30 ' EN KÜÇÜK ÖĞRENCİNİN YAŞINI VE ADINI YAZAN PROGRAM
40 S=0
50 ENK=99:TOPY=0
60 INPUT "ADI SOYADI ";AD$
70 INPUT "YAŞI ";YAS
80 S=S+1
90 IF YAS<ENK THEN ENK=YAS:AAD$=AD$
100 TOPY=TOPY+YAS
110 IF S<10 THEN 60
120 YORT=TOPY/S
130 PRINT "YAŞ ORTALAMASI";YORT
140 PRINT "EN KÜÇÜK ÖĞRENCİNİN ADI";AAD$
150 PRINT "YAŞI";ENK
160 END
```

```
10 ' BILGI YARIŞMASINA KATILAN
20 ' 5 ÖĞRENCİDEN BİRİNCİ OLAN
30 ' ÖĞRENCİNİN PUANINI VE ADINI YAZAR
40 S=0
50 ENB=0
60 INPUT "ADI SOYADI";ADŞ
70 INPUT "TOPLAM PUANI";TP
80 S=S+1
90 IF TP>ENB THEN ENB=TP:AADŞ=ADŞ
100 IF S=5 THEN 120
110 GOTO 60
120 PRINT "1. ÖĞRENCİNİN ADI";AADŞ
130 PRINT "TOPLAM PUANI";ENB
140 END
```

G. DÖNGÜ KOMUTLARI

1. ÇEVİRİM VE DÖNGÜLER

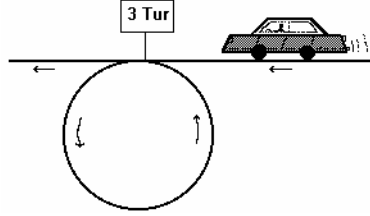
Buraya kadar gördüğümüz derslerimizde programın akışını, **GOTO** deyimiyle istediğimiz satıra gönderebiliyorduk. Döngü içerisinde belirli sayıda işlem yapılacağı zaman da, sayaçla yapıyorduk. Bunu yaparken biz bir koşul veriyor ve o koşul yerine gelince, döngüden çıkılıyordu.

a. FOR...NEXT DÖNGÜSÜ

Yazılışı:
FOR <Değişken>=X **TO** Y [**STEP**] Z
....
....
....
NEXT <Değişken>

Burada önceden belirlenmiş sayıda döngü ile program yapmayı göreceğiz. Bu sorunumuzu **FOR ... NEXT** ile çözümlenebiliriz. **FOR...NEXT** deyimleri arasına yazılan işlemleri belirlediğimiz sayıda yaparak bize bu kolaylığı sağlar.

Döngü konusunu daha iyi anlayabilmek ve pekiştirmek için bir şekilde açıklayalım:



Bu yolda seyretmekte olan bir otomobil, trafik kurallarına uymak zorundadır. Bu kurallara uymadığı takdirde, ya trafik kazası geçirecek ya da trafik kurallarına uymadığı için cezaya çarptırılacaktır. Yoldaki levhada **Üç Tur** işaretini görünce, sürücü bu dairenin içine girerek üç tur attıktan sonra yoluna devam edebilecektir. Doğaldır ki dairenin içinde giderken de uyması gereken kurallar ya da dümen kırma, yavaşlama gibi otomobili kullanmak için gerekli hareketleri yapmak zorundadır. Bu kuralları yerine getirmeden yola devam edemeyecektir. **FOR...NEXT** döngüsünü de bu olaya benzetebiliriz.

Daha önce yaptığımız **1**'den **20**'ye kadar sayıları toplayan programı **FOR...NEXT** döngüsüyle yapalım. Ama önce döngünün nasıl işlediğini ve yazılımını görelim:

FOR deyiminden sonra bir değişken belirlememiz gerekir. Sonra bu değişkenin ilk değerini vereceğiz. Burada **X** ile gösterilen değer **Başlangıç Değeri**'dir. **TO** deyiminden sonra **Bitiş Değeri** verilir. **STEP** deyiminden sonra **Artış Değeri** verilir. Bu işlem belirlendikten sonra döngü içerisinde neler yapılmasını istiyorsak bunları yazarız. Döngü **NEXT** deyiminiyle kapatılır. **NEXT** deyiminden sonra yukarıda verdiğimiz değişkeni yeniden yazarız.

```
10 FOR I=0 TO 20 STEP 5
...
...
...
90 NEXT I
```

Yukarıdaki olayın açıklaması şöyledir: **0**'dan başla **20**'ye kadar **5**'er artırarak aradaki işlemleri yap. **STEP** burada **Adım** olarak tanımlanır.

Şimdi gelelim daha önce sayaçla yaptığımız, **1**'den **20**'ye kadar sayıları toplayan programı, **FOR...NEXT** döngüsüyle yapmaya:

```
10 CLS
20 FOR S=1 TO 20
30 TOPLAM=TOPLAM+S
40 PRINT S,TOPLAM
50 NEXT S
60 END
```

Program başlarken **1**'den başlayacak **20**'ye kadar aradaki işlemleri yapacaktır. Yani **S** değeri başlangıçta **1** olacak, **TOPLAM**'a depolanacak. **S** (Sayı) ve **TOPLAM** yazılacak. **NEXT S** görüldüğünde tekrar **FOR** satırına dönecektir. Her geri dönüşte **S**'nin değeri bir artacaktır. Bunu şu şekilde de tanımlayabiliriz. **S** değeri **20** olana kadar aradaki işlemleri yap.

GOTO deyimiyile döngüye bir mesajı soktuğumuzda sonsuz döngü olmuştu.

```
10 CLS
20 PRINT "KAZIM TANIŞ"   `Programı sonsuz döngüye sokarak
30 GOTO 20               `KAZIM TANIŞ'ı sonsuza kadar yazar.
```

Bunu böyle değil de **KAZIM TANIŞ**'ı **20** kez yazmasını isteseydik sayaçla yaptığımızda şu şekilde yapabilirdik:

```
10 CLS
20 S=0
30 S=S+1
40 PRINT "KAZIM TANIŞ"
50 IF S=20 THEN 70
60 GOTO 30
70 END
```

Sayaçla yaptığımız programı şimdi **FOR...NEXT** döngüsüyle yapalım:

```
10 CLS
20 FOR S=1 TO 20
30 PRINT "KAZIM TANIŞ"
40 NEXT S
```

KAZIM TANIŞ'ı belirttiğimiz gibi **20** kez yazarak program bitecektir. **S** değeri **20** olana kadar **NEXT S**'yi görünce geri dönecek, aradaki işlemleri yapacaktır.

Bu kez **1**'den **100**'e kadar sayıları yazmasını isteyelim:

```
10 CLS
20 FOR S=1 TO 100
30 PRINT S;
40 NEXT S
50 END
```

1'den **100**'e kadar olan sayıları yanyana yazacaktır. **PRINT S** deyiminden sonra hiçbir işaret koymasaydık, bu kez alt alta yazacaktı. (;) Noktalı Virgül yerine (,) Virgül koysaydık aralıklı olarak yazacaktı.

STEP (Adım) deyimiyile bir örnek yapalım:

```
10 CLS                                RUN ↵
20 FOR I=0 TO 20 STEP 5                0
30 PRINT I                             5
40 NEXT I                              10
50 END                                  15
                                         20
                                         Ok
```

0'dan 20'ye kadar 5'er atlayarak yazacaktır.

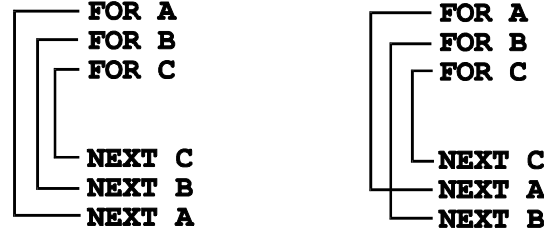
1) FOR/NEXT İçinde READ/DATA Uygulamaları

5 öğrencinin adını soyadını, iki yazılı ve bir sözlüsünü okutarak, ortalamalarını hesaplayan program:

```
10 CLS
20 PRINT "ADI SOYADI", "YAZILILAR", "SÖZLÜ", "ORTALAMA"
30 PRINT "=====", "=====", "=====", "====="
40 FOR I=1 TO 10
50 READ ADIŞ, YAZL1, YAZL2, SOZL
60 ORT=(YAZL1+YAZL2+SOZL)/3
70 PRINT ADIŞ, YAZL1; YAZL2, SOZL, ORT
80 NEXT I
90 DATA FILİZ ONACAK, 60, 50, 60
100 DATA FÜSUN ÜRGER, 55, 70, 60
110 DATA İHSAN İNCE, 60, 80, 75
120 DATA NUSRET ÇELİK, 70, 60, 85
130 DATA ZEKİ ÇATAV, 50, 70, 50
140 DATA HANDAN İLGAZ, 45, 80, 50
150 DATA SEVGI MUTLU, 60, 75, 75
160 DATA EMİNE ONACAK, 70, 60, 85
170 DATA TOLGA SEYFELİ, 75, 45, 60
180 DATA HASAN YAVUZ, 75, 40, 50
200 END
```

2) İççe Döngüler

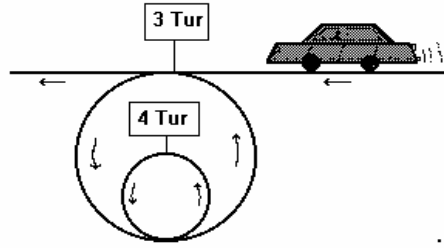
Bazı durumlarda, bir programda birden fazla döngüye gereksinim duyulabilir. Böyle durumlarda döngülerin çakışmamasına dikkat etmek gerekir. Bu konuda ilke şudur: En son açılan döngü, ilk olarak kapatılır. Böylelikle döngüler iç içe olur ve çakışmazlar.



Doğru Yazım
(Döngüler Kesişmiyor)

Yanlış Yazım
(Döngüler Kesişiyor)

Bu kez daire sayısını 2'ye çıkardık. Aşağıdaki şekil de bunu gösteriyor:



Bu yolda şoför, ilk daireye **3 tur** atmak için girecektir. Daireye girdikten sonra, bir iç daire ile karşılaşacak. Buradaki İşaret Levhası'nda **4 TUR** yazmaktadır. Şoför burada **4 tur** atacak ve tekrar dış daire içine girecektir. Ama daha dış daire için söz konusu olan **3 tur** bitmediğinden, dairenin içine tekrar girecek ve **2. tur**'a başlayacaktır. İç dairedeki 4 tur levhasıyla yeniden karşılaşacağından bu koşulu yerine getirecektir. Bu olay böyle sürüp gidecek; koşullar yerine gelince yoluna devam edebilecektir. İç içe **FOR...NEXT** döngüsü de bu olayın aynısıdır.

İç içe döngüye aşağıdaki örnekle girelim:

```
10 CLS
20 FOR A=1 TO 3
30 PRINT "DIŞ DAIRE";A;" . TUR"
40 FOR B=1 TO 4
50 PRINT "İÇ DAIRE";B;" . TUR"
60 NEXT B
70 NEXT A
80 END
```

Programı çalıştıralım:

```
RUN ↵
DIŞ DAIRE 1 . TUR
        İÇ DAIRE 1 . TUR
        İÇ DAIRE 2 . TUR
        İÇ DAIRE 3 . TUR
        İÇ DAIRE 4 . TUR
DIŞ DAIRE 2 . TUR
        İÇ DAIRE 1 . TUR
        İÇ DAIRE 2 . TUR
        İÇ DAIRE 3 . TUR
        İÇ DAIRE 4 . TUR
DIŞ DAIRE 3 . TUR
        İÇ DAIRE 1 . TUR
        İÇ DAIRE 2 . TUR
        İÇ DAIRE 3 . TUR
        İÇ DAIRE 4 . TUR
Ok
```

Sıfırdan (0) başlayıp, 100'e kadar 10'ar atlayarak giden sayıların kareleri ve küplerini alan, sayılarla birlikte ekrana yazdıran programı yapalım:

```
10 CLS
20 PRINT "SAYI", "KARESİ", "KÜP"
30 PRINT "====", "=====", "===="
40 FOR S=0 TO 100 STEP 10
50 KARE=S^2
60 KUP=S^3
70 PRINT S, KARE, KUP
80 NEXT S
90 END
```

100'den geriye doğru 0'a kadar 10'ar azalarak giden sayıların **karekök**'lerini hesaplayıp yazan program:

```
10 CLS
20 PRINT "SAYI", "KAREKÖK"
30 PRINT "====", "====="
40 FOR S=100 TO 0 STEP -10
50 KAREKOK=S^(1/2)
60 PRINT S, KAREKOK
70 NEXT S
80 END
```

Bileşik faiz hesabını yapan program:

```
10 INPUT "KAPITALI GIRINIZ";K#
20 INPUT "FAİZ FİYATINI GIRINIZ";T
30 INPUT "KAÇ YIL KALACAK";N
40 FOR S=1 TO N
50 K#=K#+(K#*T/100)
60 PRINT S; " ' INCI YIL", K#
70 NEXT S
```

Örnek: Belirli sayıda işçinin ücret bordrosunu hesaplayan program:

```
10 CLS:INPUT "KAÇ İŞÇİLİK BORDRO";N:CLS
20 PRINT "ADI SOYADI", "BRÜT ÜCRET", "S.S.K.P.", "TOP.VERGI",
"NET ÜCRET"
30 PRINT "====", "=====", "=====", "=====",
"====="
40 FOR S=1 TO N
50 INPUT "İŞÇİNİN ADI SOYADI";AD$
60 INPUT "ÇALIŞTIĞI GÜN";CG
70 INPUT "GÜNLÜK ÜCRET";GU
80 BU=CG*GU
90 SSKP=BU*.14
100 OZIND=3000*CG
110 GVM=BU-(SSKP+OZIND)
120 GV=GVM*.25
130 DV=BU*.004
140 TVERGI=GV+DV
150 NETUCR=BU-(SSKP+TVERGI)
160 PRINT AD$, BU, SSKP, VERGI, NETUCR
170 NEXT S
180 END
```

b. WHILE...WEND DÖNGÜSÜ

Öne sürülen koşul sağlanıncaya kadar aradaki işlemlerin yapılmasını sağlar.

Yazılışı:

```
WHILE <Koşul>
...
...
WEND
...
...
```

Kuralları:

1. Koşul **WHILE** deyiminden önce tanımlanmalıdır.
2. Koşulu yerine getirecek ifade, **WHILE...WEND** arasında belirtilmelidir.

Örnek:

```
10 CLS
20 S=0
30 WHILE S<>100
40 S=S+5
50 KARE=S^2
60 PRINT "SAYI";S,"SAYININ KARESİ";KARE
70 WEND
80 END
```

Yukarıdaki programda **S** sayacına ilk değer olarak **0** atadık. **WHILE** ile **S**, **100**'e eşit değilse aradaki işlemleri yapmasını istedik. **S**'yi döngü arasında **5**'er arttırdık. **S**, **100** değerini alana kadar aradaki işlemleri yaptı.

c. WHILE NOT...WEND

Önerilen koşulun değilini yapar. Bu kez de aynı işlemi olumsuz koşul öne sürerek yapalım:

```
10 CLS
20 S=0
30 WHILE NOT S=100
40 S=S+5
50 KARE=S^2
60 PRINT "SAYI";S,"SAYININ KARESİ";KARE
70 WEND
80 END
```

Burada da aynı işlemi, **S=100** değilse yap, komutuyla yaptık.

H. DİZİNLİ DEĞİŞKENLER ve BOYUT KULLANIMI

Daha önce gördüğümüz konularda değişkenlere **LET**, **READ-DATA** ve **INPUT** ile değer atanmıştı. Ancak bu komutlarla tek değer atanabiliyordu ve yeni bir değer atamasından sonra eski değer kayboluyordu. Yani girilen en son değer, bir önceki değeri ortadan kaldırıyor. Değişkenin bir önceki değerinin kullanılması olanaksızlaşıyordu. Şimdi bunu ortadan kaldıracak bir değişken göreceğiz. Buna **Dizimli (İndisli) Değişken** adı verilir. İndis numaraları değiştirilerek aynı değişkene birden çok değer atayabiliriz. Bu durumda indis numaraları değişik olacağından karışıklık da yaratılmayacaktır. İndisli değişkenlerde, değişken numarası değişken adından sonra ayrıç (..) içinde verilir. **AS(10)**, **A(5)** gibi. Bu örneklerde **AS** alfasayısal dizi değişkenine **10**, **A** sayısal dizi değişkenine de **5** değişik değer atanabileceği verilmiştir.

1. DIM

İndisli değişkenler en çok **10** değişik değer alabilirler. Bazı bilgisayarlarda bu değer **11**'e kadar çıkabilir. Bu tip bilgisayarlarda dizin **0**'dan başladığından bu fark oluşur. Bazen bu sayı yetmemektedir. Artırabilmek için Dimension deyiminden gelen **DIM** komutu kullanılır.

Yazılışı:
DIM <Değişken> (<X>), . . .

Burada **X**, o dizinin en çok kaç değer alacağını belirtir. Çok sayıda dizi değişkeni bir tek **DIM** ile tanımlamak olanaklıdır.

DIM SAYI(15),ADS(15),MAAS(15) gibi.

Tek boyutlu dizilerde her değişkene bir göz ayrılır. Değerler bu değişkene ait gözde depolanırlar. **ADS(10)**, **I(5)** her değişkene yalnızca bir çeşit değer verilebilir.

2. OPTION BASE

İndis değerinin **sıfırdan** değil de **1**'den başlaması isteniyorsa **OPTION BASE** deyimi kullanılır.

Yazılışı:
OPTION BASE 1

a. TEK BOYUTLU DİZİLER

Tek boyutlu dizilerde, indisli değişkene, yalnızca bir satırlık atama yapılabilir.

1) Sayısal Dizi Açma

Sayısal dizi açmak demek, DIM ile belirlenen boyuta sayı atamak demektir.

DIM MAAS (10) ,DIM VERGI (10) gibi...

Aşağıda sayısal diziye bir örnek verilmiştir:

```
5 CLS
10 OPTION BASE 1
20 DIM NT(10)
30 FOR N=1 TO 10
40 INPUT "ALDIĞI NOT" ;NT(N)
50 NEXT N
60 PRINT "SIRA NO" , "ALDIĞI NOT"
70 PRINT "=====", "====="
80 FOR N=1 TO 10
90 PRINT N ,NT(N)
100 NEXT N
110 END
```

Bu program 10 öğrencinin sıra numarası ve aldığı notunu alt alta yazar.

2) Alfasayısal Dizi Açma

Sayısal değişkenli diziler için verdiğimiz özellikler alfasayısal değişkenli diziler için de geçerlidir.

DIM CINS\$ (12) ,AD\$ (12) gibi...

Burada da tanımladığımız boyuta alfasayısal ifade atayabiliriz. Alfasayısal ifade atamak için değişkenin sonuna \$ işaretini yazmamız gerekir.

```
10 CLS
20 DIM ADI$ (5)
30 FOR I=1 TO 5
40 READ ADI$ (I)
50 NEXT I
60 PRINT "ADI"
```

```
70 PRINT "==="
80 FOR I=1 TO 5
90 PRINT ADIŞ(I)
100 NEXT I
200 DATA AYŞE , FATMA , ALI , FILİZ , METİN

10 ' 10 ÖĞRENCİNİN ADI SOYADI VE ORTALAMASI
20 ' BİR DİZGİ İÇERİSİNDE OKUNMAKTA VE
30 ' SINIFI GEÇENLERİN (50) LİSTESİNİ VERMEKTEDİR
40 DIM ADŞ(10) ,ORT(10)
50 FOR S=1 TO 10
60 INPUT "ADI SOYADI";ADŞ(S)
70 INPUT "ORTALAMA";ORT(S)
80 NEXT S
90 CLS
100 PRINT "SINIFI GEÇENLERİN LİSTESİDİR"
110 PRINT "SIRA NO" , "ADI SOYADI" , "NOTU"
120 FOR S=1 TO 10
130 IF ORT(S)>=50 THEN PRINT S ,ADŞ(S) ,ORT(S)
140 NEXT S
150 END
```

b. ÇİFT BOYUTLU DİZİLER

Bu dizilerde, daha çok göz açarak, daha çok değer ya da ifadeyi bellekte tutabiliriz. Yani matris kullanımı söz konusudur. Şöyle ki; **A(10,5)** şeklinde bir göz açtığımızda; **10** satırı, **5** sütunu belirler. Aşağıdaki çizelgede bu konuyu daha iyi anlayabileceğiz. **A(4,3)**'ün çizimini verelim:

	1	2	3	S	
1	A(1,1)	A(1,2)	A(1,3)	A	
2	A(2,1)	A(2,2)	A(2,3)	T	
3	A(3,1)	A(3,2)	A(3,3)	I	
4	A(4,1)	A(4,2)	A(4,3)	R	
	S	Ü	T	Ü	N

Bu tabloda da gördüğümüz gibi **12** değişik değeri, çift boyutlu olarak bir diziye depolayabiliyoruz.

1) Sayısal Dizi Açma

Tek boyutluda olduğu gibi, sayısal değer atanacağı için sayısal değişken kullanılmalıdır. Parantez içerisinde boyut belirlenmelidir.

DIM SAYI (10,5) gibi...

Aşağıda (3,2) boyutunda sayısal örnek verilmiştir.

```
10 CLS
20 DIM S(3,2)
30 FOR K=1 TO 3
40 FOR L=1 TO 2
50 READ S(K,L)
60 NEXT L
70 NEXT K
80 DATA 1991,50,1992,70,1993,45
90 PRINT "YILI","TON"
100 PRINT "====","===="
110 FOR K=1 TO 3
120 FOR L=1 TO 2
130 PRINT S(K,L),
140 NEXT L
150 PRINT
160 NEXT K
```

2) Alfasayısal Dizi Açma

Alfasayısal çift boyutlu dizilerde, değişkenin sonuna \$ işareti konur. Parantez içerisinde boyutlar belirlenir.

DIM ADI\$(5,4) gibi...

Çift boyutlu diziye örnek olarak, (3,2) boyutlarında bir dizi açarak öğrencilerin adını soyadını ve notlarını atayan bir program yapalım:

```
10 CLS
20 DIM A$(3,2)
30 FOR K=1 TO 3
40 FOR L=1 TO 2
50 READ A$(K,L)
60 NEXT L
70 NEXT K
```

```
80 DATA UMUTCAN TANIŞ,50,FILİZ ONACAK,70,AYŞE ABAT,45
90 PRINT "ADI SOYADI", "NOTU"
100 PRINT "=====", "====="
110 FOR K=1 TO 3
120 FOR L=1 TO 2
130 PRINT AŞ (K,L) ,
140 NEXT L
150 PRINT
160 NEXT K
```

3. BELLEK ve BOYUT SIFIRLAMA

a. CLEAR

Yazılışı:
CLEAR

Bu komutla bellekteki tüm değişkenler sıfırlanır. Bir programı çalıştırdıktan sonra bellekte değişkenlerin değerleri kalır. Programı kesme, döngü vb. durumlarda, yarıdan çalıştırdığımızda değişkenler kaldığı değerlerden devam edebilirler. Böyle bir programda geri dönüşlerin olduğu satırda **CLEAR** komutu kullanılırsa, bütün değişkenlerin değerleri sıfırlanacağından yeniden çalıştırma durumunda, istenilmeyen sonuçların oluşması engellenir. Bu komutla sayısal değişkenler sıfırlanırken, alfasayısal değişkenler de boşalır.

b. ERASE

Yazılışı:
ERASE

ERASE ile **DIM** kullanırken boyutlara atadığımız değerleri sıfırlarız.

I. HAZIR FONKSİYONLAR

Basic'de bazı işlemlerin yapılabilmesi için hazır komut ve deyimler vardır. Bu bölümde bu komutları göreceğiz. Logaritma aldırma, sayıları yuvarlama, tarih ve saat yazdırma gibi...

1. MATEMATİKSEL FONKSİYONLAR

a. INT

Yazılışı:
INT (X)

X sayısından, küçük veya ona en çok eşit, en büyük tam sayı değerini bulur.

```
PRINT INT (2 . 78) , INT (2 . 25)
      2                2
```

```
PRINT INT (-2 . 78) , INT (-2 . 25)
      -3                -3
```

b. FIX

Yazılışı:
FIX (X)

X kesirli sayısının, **tam sayı kısmını** verir. Sayının negatif ya da pozitif olması önemli değildir. Tam sayıyı yazarak, kalan bölümü atar.

```
PRINT FIX (3 . 25) , FIX (3 . 88)
      3                3
```

```
PRINT FIX (-3 . 25) , FIX (-3 . 88)
      -3                -3
```

c. CINT

Yazılışı:
CINT (X)

X sayısını **tamsayıya** (integer) çevirir. Sayı **-32768** ile **+32767** arasında olmalıdır.

```
PRINT CINT (478 . 75)
      479
```

d. LOG

Yazılışı:
LOG (X)

Logaritma. X sayısının **doğal logaritmasını** bulur.

```
PRINT LOG (3) , LOG (453)
1.098612      6.115892
```

e. SIN

Yazılışı:
SIN (X)

X açısının, **Radyan** değeri olmak üzere **Sinüs**'ünü bulur.

```
PRINT SIN (22)
- .7509873
```

f. COS

Yazılışı:
COS (X)

Radyan cinsinden verilmiş, X değerinin **Kosinüs**ünü bulur.

```
PRINT COS (180)
- .5984601
```

g. SQR

Yazılışı:
SQR (X)

X değerinin **karekökünü** bulur.

```
PRINT SQR (81) , SQR (5)
9      2.236068
```

h. TAN

Yazılışı:
TAN (X)

X açısının, **Radyan** değeri olmak üzere **Tanjantını** bulur.

PRINT TAN (2)
-2.18504

i. ATN

Yazılışı:
ATN (X)

X'in **arktanjat** değerini bulur.

PRINT ATN (234)
1.566523

j. ABS

Yazılışı:
ABS= (X)

X değişkeninin mutlak değerini verir.

10 PRINT ABS (200-500)
20 PRINT ABS (-1250)

RUN
300

1250 İşlem ve yazdırmaların mutlak değerlerini verdi.

k. CDBL

Yazılışı:
CDBL (X)

X sayısını **çift duyarlıklı** sayıya çevirir.

PRINT CDBL (345*3)
1035

I. CSNG

Yazılışı:
CSNG (X)

X sayısını **tek duyarlıklı sayıya** çevirir.
PRINT CSNG (416.48373734#)
416.4837

m. EXP

Yazılışı:
EXP (X)

Exponential=Üstel anlamındadır. E sayısının X. kuvvetini verir. **E=2.718281**'dir.

PRINT EXP (2) PRINT EXP (6)
7.389056 403.4287

n. SGN

Yazılışı:
SGN (X)

X'in aldığı değere göre **1, 0, -1** elde ederek döner.

X>0 ise SGN(X)=1
X=0 ise SGN(X)=0
X<0 ise SGN(X)=-1 olur.

PRINT SGN (20) , SGN (0) , SGN (-20)
1 0 -1

o. RND

Yazılışı:
RND [(X)]

RND, 0 ile 1 arasında rastgele sayı üretir. Bu sayılar aynı sırayı takip ederler. Hep aynı sayının gelmemesi için **RANDOMIZE** komutu kullanılır.

PRINT RND ↵
.1213501


```
10 RANDOMIZE TIMER      Burada RANDOMIZE TIMER ile, bilgisayarın
20 A=RND*1              belleğinde bulunan saate ayarlı olarak sayı üretildi.
30 PRINT A
```

Rastgele tam sayı üretmek istiyorsak, şu şekilde kullanırız.
A=INT ((RND*n) +1)

n yerine yazılacak sayıya kadar sayılar üretir. **30** verecek olursak **1-30** arasında rastgele sayı üretilir.

2. KARAKTER FONKSİYONLAR

a. CHR\$

Yazılışı:
CHR\$ (n)

ASCII kodu verilen sayısal değerın karakter karşılığını verir. **n**'in değeri **0-255** arasında bir sayıdır.

```
PRINT CHR$ (65) ,CHR$ (97) ,CHR$ (128) ,CHR$ (40)
A                a                Ç                <
```

b. STRINGS

Yazılışı:
STRING\$ (n,m)
veya
STRING\$ (n,X\$)

Bu deyim **n** kadar, **m** ASCII kodunun karakter karşılığını verir. **X\$** karakter dizisinin ilk karakterini **n** kadar verir.

```
10 PRINT STRING$ (10,61)
20 PRINT STRING$ (10,65)
30 PRINT STRING$ (10,"A")
40 PRINT STRING$ (10,"=")
```

RUN ↵

```
=====
AAAAAAAAAA
AAAAAAAAAA
=====
```

Belirtilen sayı kadar, belirtilen karakteri yazdı.

c. LEN

Yazılışı:
LEN (X\$)

X\$ değişkeninin kaç karakter olduğunu, yani uzunluğunu verir.

```
10 AD$="ORAL C. SEKENDUR"  
20 C=LEN(AD$)  
30 PRINT C
```

```
RUN ↵  
16          Yazılan karakter dizisinin uzunluğunu verdi.
```

d. LEFT\$

Yazılışı:
LEFT\$ (X\$, n)

Bu deyimle X\$ değişkeninin, soldan itibaren n kadar karakterini alır.

```
10 X$="BILGISAYAR"  
20 C$=LEFT$(X$, 5)  
30 PRINT C$
```

```
RUN ↵          Soldan itibaren BİLGİSAYAR sözcüğünün ilk 5 karakterini okudu ve  
BILGI          bunu yazdı.
```

e. MID\$

Yazılışı:
MID\$ (X\$, n [, m])

Bu deyim X\$ karakter dizisinin n. karakterinden başlayarak, m kadarını alır.

```
10 X$="BILGISAYAR"  
20 C$=MID(X$, 2, 4)  
30 PRINT C$
```

```
RUN ↵          BİLGİSAYAR sözcüğünün 2. karakterinden başlayarak 4 karakterini  
ILGI          aldı.
```

f. RIGHTS

Yazılışı:
RIGHT\$ (X\$, n)

X\$ değişkeninin sağdan itibaren, n karakterini alır.

```
10 X$="BILGISAYAR"  
20 Y$=RIGHT$(X$, 3)  
30 PRINT Y$
```

```
RUN ↵  
YAR          BİLGİSAYAR sözcüğünün sağdan itibaren, 3 karakterini aldı.
```

g. SPACES

Yazılışı:
SPACE\$(n)

Bu deyim n sayısı kadar boşluk verir.

```
10 A$=SPACE$(10)  
20 X$="BİLGİSAYAR"  
30 PRINT A$;X$
```

```
RUN ↵          10 karakterlik boşluk bırakarak BİLGİSAYAR  
                BILGISAYAR sözcüğünü yazdı.
```

h. VAL

Yazılışı:
VAL (X\$)

X\$ alfasayısal değişkeninin değerini sayısalaya çevirir. STR\$ fonksiyonunun karşıtıdır. Çevrilmek istenen karakter dizinin ilk karakteri sayı değilse, 0 değeri verilir.

```
10 PRINT VAL("05/23/1920")  
20 PRINT VAL("BILGISAYAR 3 ADET")
```

```
RUN ↵  
5  
0
```

Bir örnek daha yapalım:

```
10 INPUT X$
20 S=VAL(S$)
30 PRINT S
```

Girilen karakter bilgiyi sayısalaya dönüştürür.

i. STR\$

Yazılışı:
STR\$(n)

n sayısal değerini karakter bilgiye dönüştürür. VAL fonksiyonunun tersini yapar.

```
10 N=15765
20 X$=STR$(N)
30 PRINT LEN(X$)
```

```
RUN ↵
6
```

j. SWAP

Yazılışı:
SWAP <1. Değişken>, <2. Değişken>

Değişkenlerin değerlerini birbiri ile değiştirir.

```
10 A=20 : B=40
20 SOZ1$="MERHABA" : SOZ2$="ARKADAŞIM"
30 PRINT A, B
40 PRINT SOZ1$, SOZ2$
50 PRINT "DEĞİŞİM======"
60 SWAP A, B
70 SWAP SOZ1$, SOZ2$
80 PRINT A, B
90 PRINT SOZ1$, SOZ2$
100 END
```

```
RUN ↓
 20          40
MERHABA     ARKADAŞIM
DEĞİŞİM=====
 40          20
ARKADAŞIM   MERHABA
Ok
```

k. INSTR

Yazılışı:
INSTR ([n,]X\$, Y\$)

Bu deyim bir karakter dizisinde, bir karakter dizisini aramak ve konumunu göstermek için kullanılır.

```
10 AD$="BILGISAYAR"
20 X$="S"
30 I=INSTR(AD$,X$)
40 PRINT AD$
50 PRINT "ARADIĞINIZ HARF:";X$
60 PRINT "KONUMU:";I
```

Bu programı çalıştırdığımızda, **40.** satırla karakter dizisini yazacak; **50.** satırda aranan harfi verecek; **60.** satırda konumunu verecektir.

Bu sonucu **30.** satırda **INSTR** fonksiyonunu kullanarak başardık. **AD\$** dizisi içinde **X\$** dizisini (karakterini) aradık.

```
RUN
BILGISAYAR
ARADIĞINIZ HARF: S
KONUMU: 6
```

l. ASC

Yazılışı:
ASC (x\$)

Verilen karakterin **ASCII** kodunu elde etmek için kullanılır. Eğer dizi birden çok karakterli ise verilen dizinin ilk karakterinin, **ASCII** kodunu verir. Kitabın sonunda karakterlerin **ASCII** ve **KARAKTER** kodları ile **HEXADECIMAL** karşılıkları verilmiştir.

```
PRINT ASC ("A" ) ,ASC ("AYHAN" ) ,ASC ("a" )
      65           65           97
```

m. HEX\$

Yazılışı:
HEX\$ (n)

Bu fonksiyon, onlu sistemde verilen bir sayının onaltılık karşılığını verir.

```
PRINT HEX$ (188) ,HEX$ (493)
      BC           1ED
```

n. OCT\$

Yazılışı:
OCT\$ (n)

10'luk sistemdeki sayıyı, 8'lik sisteme çevirir.

```
10 X=845
20 Y$=OCT$ (X)
30 PRINT Y$
```

```
RUN
1515           845 onluk sayısını, 8'lik sayıya çevirdi.
```

3. İMLEÇ HAREKET FONKSİYONLARI

Bu fonksiyonlar imlecin durumunu değiştirir, ekrandaki konumunu belirler.

CSRLIN DEĞİŞKENİ ve POS FONKSİYONU

Yazılışı:
Y=CSRLIN
Z=POS (n)

Bu deyimler imlecin yerini (pozisyonunu) belirler. **CSRLIN** satırı, **POS(n)** o satırda kaçınıcı karakterde olduğunu belirler.

Satır=CSRLIN Bu şekilde kullandığımızda, **PRINT** deyimiyile yazmasını iste-
Sütun=POS(0) seydik, imlecin o anda bulunduğu konumu belirleyecekti.

4. KARAKTER VERİ GİRİŞ FONKSİYONLARI

Karakter girişi sağlayan fonksiyonlardır.

a. INKEY\$

Yazılışı:
X\$=INKEY\$

Bu deyim klavyeden girilen bir karakteri okur ve belleğe aktarır.

200 A\$=INKEY\$:A\$="E" THEN 10 Girilen E ise **10.** satıra gider.

veya

320 INKEY\$="" THEN 320 Herhangi bir tuşa dokununcaya kadar **320.** satırda bekler.

b. INPUT\$

Yazılışı:
INPUT\$(X)

Klavyeden girilen, **X** kadar karakteri okur ve belleğe aktarır. Okuduğu karakterler ekranda görüntülenmez.

```
10 CLS
20 PRINT "ŞİFREYİ GIRINIZ";
30 SF$=INPUT$(5)
40 IF SF$="GÜVEN" THEN 50 ELSE PRINT "BİLEMEDİNİZ":GOTO 20
50 PRINT "TEBRİKLER BILDİNİZ. DEVAM EDEBİLİRSİNİZ."
```

c. LINE INPUT

Yazılışı:

LINE INPUT <"Mesaj">,<Değişken>

255 karaktere kadar olan karakter dizisini, okuyarak tek string göze yerleştirir. Daha önce **INPUT** deyimiyle bazı karakterleri giremiyorduk, Örneğin , (virgül) gibi. Bu deyimle (,) dahil benzer karakterleri girebiliriz. Tek bir satırda birkaç cümleciciği girmek istediğimiz zaman kullanılır.

LINE INPUT "ADRESI GIR (MAHALLE, CADDE, SOKAK, NO) ";ADR\$

Bilgiler aralara virgül de konarak girilir.

5. TARİH ve SAAT FONKSİYONLARI

a. DATES

Yazılışı:

DATE\$=X\$ Deyim olarak yazılışı.

Y\$=DATE\$ Değişken olarak yazılışı.

Bu deyim bilgisayarın belleğinde bulunan tarihi okumak ya da değişken olarak tarih atamada kullanılır.

AA-GG-YY ya da **AA/GG/YY** şeklinde, yani **ay-gün-yıl** yazılımı sağlar.

PRINT DATE\$ ↵

03-09-1990

Bilgisayarın belleğinde bulunan tarihi görüntüler.

b. TIMES

Yazılışı:

X\$=TIME\$

Bilgisayarın belleğinde bulunan zamanı okutmak ya da belirtilen zamanı bir değişkene atamak için kullanılır.

SS:DD:SN şeklinde giriş yapılır. **SS**=Saat, **DD**=Dakika, **SN**=Saniye'yi tanımlar. Yalnızca saati yazacak olursak diğer değerler **0** kabul edilir.

PRINT TIME\$ ↵

18:49:00

6. GRAFİK FONKSİYONLARI

a. SCREEN

Yazılışı:

SCREEN [**Mod**] Deyim olarak

veya

SCREEN [**Satır**,**Sütun**] Fonksiyon olarak

Birinci yazılışta grafik moduna girmek veya çıkmak için kullanılır.

Mod: 0 ve 1 modu ekranın 40 sütunluk olduğu durumlar için, 2 modu ise 80 sütunluk durumlar için geçerlidir.

İkinci yazılışta, fonksiyon olarak kullanılır.

Satır: 1-25 arasında değerler alır.

Sütun: 1-40 veya 80 arasında değerler alır.

b. WIDTH

Yazılışı:

WIDTH [**Sütun No**]

Daha önce de gördüğümüz bu komutu grafik ve çizim sırasında yine ekranı bölmek için kullanacağız.

Sütun No: 1-40 ve 80 arası değerlerdir.

WIDTH 40 Ekranı 40 sütunluk yazı için ayırır.

c. WIEW

Yazılışı:

WIEW [**SCREEN**] [(**x1**,**y1**) - (**x2**,**y2**) , [**Renk**] , [**Kenar Rengi**]

x1,y1 x,y koordinatlarının sol üst noktasını belirtir.

x2,y2 x,y koordinatlarının sağ alt noktasını belirtir.

Renk Tanımlanan bölgenin rengi.

Kenar Rengi: Tanımlanan bölgenin kenar rengi.

WIEW, **SCREEN** ile birlikte kullanılırsa mutlak koordinatlara göre çizim yapılır. Bu durumda **WIEW** koordinatlarına göre çizim yapılamaz.

d. WIEW PRINT

Yazılışı:
WIEW PRINT [İlk Sınır] TO [Son Sınır]

Belirtilen satır alanını kapatarak ekranın kullanılan alanını sınırlar.

WIEW PRINT 8 TO 17

Ekranın 8. satır ile 17. satırları arasının kullanımını sağlar. Diğer bölgeleri kapatır.

e. WINDOW

Yazılışı:
WINDOW (x1, y1) - (-x2, -y2)

Grafik ekranının koordinat sistemini istediğimiz biçimde ayarlamaya yarar.

WINDOW (100, 100) - (-100, -100)

Grafik ekranını 100 birimlik ölçülere göre ayarlar. Yani bu durumda x ve y en çok 100 değerini alır.

f. LINE

Yazılışı:
LINE (x1, y1) - (x2, y2) , Renk, B [F]

Ekranın x1,y1 noktasından x2,y2 noktasına çizgi çizdirilir veya bu noktalar arasında kutu yapılır.

(x1,y1)-(x2,y2) noktalarına çizgi çizilebildiği gibi bu koordinatlar bir dikdörtgenin karşılıklı köşelerini tanımlar.

Renk: Çizginin rengi
B Kutu çizdirmek için
F Kutunun içi doldurulacağı zaman, **B** ile birlikte kullanılır.

```
5 ' LINE için örnek program
10 CLS:KEY OFF
20 SCREEN 2
30 WINDOW (-100,-100)-(100,100)
40 LINE (-30,-30)-(30,30),7,BF
50 LINE (-75,-75)-(75,75),7,B
60 LINE (-90,-90)-(-60,-50),7,B
70 LINE (-90,90)-(-60,50),7,B
80 LINE (90,90)-(60,50),3,B
90 LINE (90,-90)-(60,-50),1,B
100 LINE (-90,-90)-(90,90),7
110 LINE (-90,90)-(90,-90),7
120 END
```

g. CIRCLE

Yazılışı:

CIRCLE (x,y),r,Renk,[Başlama,Bitiş]

Daire çizdirmek için kullanılır. Belirtilen koordinatı merkez olmak ve yarıçapı belirtilmek üzere daire çizer.

x,y	Dairenin merkez koordinatı
r	Dairenin yarıçapı
Renk	Dairenin rengi
Başlama,Bitiş	Çizilecek yayın başlama ve bitiş açısı (Radyan cinsinden)

CIRCLE (0,0),40,7

0,0 orijininden başlayan, yarıçapı **40 birim** ve **7 renginde** daire çizer.

```
5 CLS
10 SCREEN 2
20 WINDOW (-100,-100)-(100,100)
30 CIRCLE (0,0),15
40 CIRCLE (50,50),15
50 CIRCLE (-50,-50),15
60 END
```

Ekranın ortasına ve çapraz iki köşesine daire çizer.

```
5 CLS:KEY OFF
10 SCREEN 2
20 WINDOW (100,100)-(-100,-100)
30 FOR D=0 TO 360 STEP 10
40 DRAW "TA=D;NU100C14"
45 DRAW "U50R50D50L50"
50 NEXT
55 FOR A=46 TO 75 STEP 7
60 CIRCLE (0,0),A,7
70 NEXT A
```

Ekrana oya biçiminde teker çizer.

h. DRAW

Yazılışı:
DRAW Dizi/Ifade

Bir dizi içerisinde verilen çizgileri çizer. Dizi ile belirtilen yönde çizginin hareketi sağlanır.

Aşağıdaki kurallara göre işler:

<u>Komut</u>	<u>Yön</u>
U(n)	n birim yukarı git
D(n)	n birim aşağı git
L(n)	n birim sola git
R(n)	n birim sağa git
E(n)	n birim üste sağa git
F(n)	n birim aşağı sağa git
G(n)	n birim aşağı sola git
H(n)	n birim üste sola git

- B** Gidilen yere iz bırakmadan gidilmesini sağlar.
- N** Noktanın konumunu, şekil yapıldıktan sonra eski konumuna getirir.
- Mx,y** x'in önüne + ya da - yazılarak, belirtilen noktadan itibaren + ya da -'ye göre hareket sağlar.
- An** Ekranın dönme açısını belirler. **0-3** arasında değer alır.
- n=0** ise açı **0** derece
- n=1** ise açı **90** derece
- n=2** ise açı **180** derece
- n=3** ise açı **270** derece

TAn	n'e açı olarak verilecek değere göre dönme sağlar. -360 ile +360 'a kadar değer verilir. n, - ise saat yönünde + ise ters yönde dönme sağlar.
Cn	Renk verme, COLOR deyiminde kullanılan ölçüler geçerlidir.
Sn	1'den 255 'e kadar faktör değeri alır. U,D,L,R,E,F,G,H,M komutları ile kullanıldığında çarpım olarak görev yapar. Değer verilmezse n, 4 kabul edilir. n, 4'e bölünerek görev yapar.
Xs,e	Ardarda karakter bilgilerin uygulanması. Komutun bir dizisi bir karakter bilgi dizisidir.
Ps,r	r sınır renginde çizilen şeklin içini s renginde boyar. s ve r renk için kodlardır.

Örnek:

```
10 SCREEN 2
20 DRAW "U20R20D20L20"
```

Ekrana bir dikdörtgen çizer.

```
10 CLS:KEY OFF
30 SCREEN 2
40 FOR S=0 TO 360 STEP 60
50 DRAW "TA=S;U50L50D50"
60 NEXT S
```

i. PSET

Yazılışı:

PSET [x,y] [,Renk]

Bir noktanın grafik modunda çizimini sağlar. x ve y ekranın çözünümlüne göre ve grafik moduna göre değişiklik sağlar. Renk, 0 ile 3 arası değerdir.

<u>Mod</u>	<u>x'in Değeri</u>	<u>y'nin Değeri</u>
SCREEN 1	0-319	0-199
SCREEN 2	0-639	0-199

```
5 ' PSET için örnek program
10 CLS
20 SCREEN 2
30 WINDOW (-150,-150)-(150,150)
40 FOR S=1 TO 120
50 PSET (S,S),5
60 NEXT S
70 FOR S=1 TO 120
80 PSET STEP(-1,-1),0
90 NEXT S
```

j. PRESET

Yazılışı:
PRESET (x,y) [,Renk]
PRESET deyiimiyle x ve y koordinatlarının rengi tanımlanır.

```
5 'PRESET için örnek program
10 CLS:KEY OFF
20 SCREEN 2
30 WINDOW (-100,-100)-(100,100)
40 FOR S=0 TO 100
50 PRESET (S,S),7
60 NEXT S
70 FOR S=0 TO 100
80 PRESET STEP (-1,-1)
90 NEXT S
```

k. COLOR

Yazılışı:
Yazılışı grafik moduna göre değişiklik sağlar.

<u>Mod</u>	<u>Yazılışı</u>
SCREEN 0	COLOR [Ön zemin],[Arka zemin],[Kenar Rengi]
SCREEN 1	COLOR [Ön zemin],[Palet]
SCREEN 7-10	COLOR [Ön zemin],[Arka zemin]

Önzemin 0-31 arasında ve tam sayı olmalıdır. Renkli ekranlarda 0-15 arasında bir sayıdır. Yanıp sönen bir görüntü elde edilmek isteniyorsa 16 eklenmelidir.

Arka zemin 0-7 arasında seçilecek renk, tam sayı olmalıdır.
Kenar Rengi Seçilecek renk 0-15 arasında tam sayı olmalıdır.

I. POINT

Yazılışı:
POINT (x,y)
veya
POINT (Fonksiyon)

Tanımlanan noktanın rengine dönüşü sağlar.

```
10 SCREEN 2
20 IF POINT (S,S)<>0 THEN PRESET (S,S) ELSE PSET (S,S)
30 PSET (S,S),1-POINT (S,S)
```

m. PAINT

Yazılışı:
PAINT (x,y) (İç Rengi) [,Sınır Rengi]

Sınırlanmış bir alanı boyar.

x,y	Boyamaya başlama pozisyonu
İç Rengi	Sınırlanmış şeklin içinin rengi
Sınır Rengi	Şeklin sınır rengi

n. GET (Grafik)

Yazılışı:
GET (x1,y1)-(x2,y2),Dizi Adı

Ekrandan görüntü taşıma görevini yapar. Ekrandaki görüntüyü **x1,y1** ve **x2,y2** pozisyonlarına taşır.

```
GET (x1,y1)-(x2,y2),AD
```

Ekran noktalarını **AD** dizisine okur.

o. PUT (Grafik)

Yazılışı:
PUT (x,y),Dizi,Pozisyon

x,y noktasından başlayan, dizi ile saklanan pozisyon'un rengini verir.

7. SES FONKSİYONLARI

Basic'le program yaparken, ses eklemek ya da müzik yapmak gerekebilir. Bu bölümde vereceğimiz fonksiyonlar seslendirme fonksiyonlarıdır.

a. PLAY

Yazılışı:

PLAY <"Alfabetik İfade">

Nota ile tanımlanan alfabetik ifadeye göre müzik üretir. Bu deyimle birlikte kullanacağımız, sembolleri aşağıda göreceğiz:

Notalar:

<u>Nota</u>	<u>Karşılık Gelen Sembol</u>
Do	C
Re	D
Mi	E
Fa	F
Sol	G
La	A
Si	B
Do	C

Notanın arkasına # ya da + konursa sert, yani o notanın frekansı yarım nota kadar yükselir; - konursa perdeli, yani notanın frekansı yarım nota kadar azalır.

Ton:

O(n) Oktavı belirler. Geçerli oktav **0**'dir. **0-6** arası değerler alır.
N(n) Oktav belirler. **n** notasını çaldırır. **0-84** arası değerler alır. Geçerli oktav **7**'dir.

Süre:

L(n) Nota uzunluğu belirtilir. **L1** notanın kendisi, **L4** çeyrek notadır. **1-64** arasında değer alır.
MN Normal müzik üretmek için. Nota uzunluğu **7/8**'dir.
ML Legato müzik için. Notalar dört dörtlüktür. Yani tam uzunluktadır.
MS Staccato müzik için. Notalar üç çeyrek uzunlukta çalınır.

Tempo:

P(n)	Durak. 1-64 arasında değer alır.
T(n)	Vuruş. Dakikada çeyrek notanın (L4) vuruş sayısı. 32-255 arası değer alır.
Hareket:	PLAY ve SOUND önzeminde çalar.
MF	Önzemin müziği.
MB	Arkazemin müziği. 32 nota bir zaman aralığında çalınabilir.

```
10 CLS
20 LOCATE 6,35:PRINT "HAVADA BULUT YOK"
30 PLAY"O2L8FL4GAL3AL4AL3AL4AL3GL8GP8L8CL4O3CO2GL8A
FAL3GP8"
40 PLAY"O3L8FL4GAL3AL4AL3AL4AL3GL8GP8L8CL4O4CO3GL8A
FAL3GP8"
50 PLAY"O2L8AAGGFL4AL8GGP8L8GGFFEL4GL8EFP8L8FFEEDL
4FL8GEDP8L8FFGEFL3DDP8"
60 PLAY"O3L8AAGGFL4AL8GGP8L8GGFFEL4GL8EFP8L8FFEEDL
4FL8GEDP8L8FFGEFL3DDP8"
70 END
```

b. SOUND

Yazılışı:
SOUND <Frekans>,<Süre>

Belirtilen frekans ve sürelerde ses üretir. **Frekans 37-32767** arasındadır. **Süre 0-65536** arasındadır.

SOUND 1111,.1 gibi...

8. EKLAN DÜZENLEME FONKSİYONLARI

Daha önceki bölümlerde ekran düzenleme komutlarını görmüştük. Bu bölümde biraz daha detaylı bilgiler öğreneceğiz.

a. WIDTH

Yazılışı:
WIDTH [LPRINT] <Tamsayı İfade>
WIDTH <"Donanım">,<Karakter Sayısı>
WIDTH <Dosya Numarası>,<Karakter Sayısı>

Ekran veya yazıcıdaki satıra düşen karakter sayısını ayarlamak için kullanılır.

WIDTH 40 Ekranı 40 karakterlik yapar.
WIDTH "LPT1:" ,132 Yazıcıyı 132 karakterlik yapar.

```
10 WIDTH "LPT1:" ,40
20 LPRINT "YAZICIYI BİR SATIRA EN ÇOK 40 KARAKTER YAZACAK
ŞEKİLDE AYARLAR"
30 NOŞ="38746534253641"
40 LPRINT NOŞ
50 WIDTH 80:END
```

Çıktı:

```
YAZICIYI BİR SATIRA EN ÇOK 40 KARAKTER Y
AZACAK ŞEKİLDE AYARLAR
38746534253641
Ok
```

b. COLOR

Yazılışı:
COLOR [Karakter Rengi No][,Zemin Renk No]
[,Çerçeve Rengi No]

COLOR deyimiyle, ekranın renk düzenlemesini yapabiliriz. Karakterleri, zemini ve çerçeve rengini belirleyerek, ekran görünümünün göze daha hoş gelmesini ve gözle seçimini sağlayabiliriz.

COLOR Deyiminin Renkli Ekran İçin Renk Parametreleri:

<u>RENK NO</u>	<u>RENK</u>
0	Siyah
1	Mavi
2	Yeşil
3	Gece Mavisi (Sarı-Yeşil Arası)
4	Kırmızı
5	Mor
6	Kahverengi
7	Beyaz
8	Gri
9	Açık Mavi
10	Açık Yeşil

11	Açık Gece Mavisi
12	Açık Kırmızı (Pembe)
13	Koyu Kırmızı (Eflatun)
14	Sarı
15	Parlak Beyaz
16-31	0-15 arası renklerin, yanıp sönererek görüntülenmesini sağlar.

Siyah-Beyaz (Monochrome) Ekranlar İçin Renk Kodları:

<u>RENK NO</u>	<u>RENK</u>
0	Siyah
1	Beyaz (Altı Çizili)
2-7	Yeşil veya Amber Kalem Rengi

Bu sayılara **8** ekleyerek, bu renklerin daha parlaklığını elde edebiliriz. **16** ekleyerek bu renklerin yanıp sönenlerini elde edebiliriz. Aynı kural Renkli Ekran numaraları için **0-15** arası renkleri, **16** ekleyerek yanıp söner duruma getirmek için geçerlidir.

COLOR 7,0	Siyah zemin üzerine, beyaz karakter.
COLOR 0,7	Beyaz zemin üzerine, siyah karakter.
COLOR 15,0	Parlak görüntü, siyah zemin, beyaz karakter.
COLOR 1,0	Altı çizili, siyah üzerine beyaz.
COLOR 0,0	Karakter görülmez, siyah üzerine siyah.
COLOR 31,0	Yanıp sönen, parlak beyaz görüntü, siyah zemin üzerine.

c. LOCATE

Bu deyiimi daha önce görmüştük, burada daha ileri kullanımı üzerinde duracağız.

Yazılışı:

LOCATE [**Satır No**] [**Sütun No**] [, [**İmleç**]
[**Başlama No**] [**Bitiş No**]]

Satır No	Daha önce de belirttiğimiz gibi 1-25 arası bir numaradır. Kaçıncı satıra gidileceğini belirtir.
Sütun No	Bu değer 1-40 veya 1-80 arasında olabilir. Satırın kaçınıcı karakterinde olduğunu belirtmek için kullanılır.
İmleç	0 veya 1 değerini alır. 0 verilirse imleç ekranda görüntülenmez. Verilmezse imleç ekranda yanar söner şekildedir.
BaşlamaNo	0-31 değerlerini alır. İmlecin yatay ve dikey tarama yapmasını sağlar.
Bitiş No	0-31 değerlerini alır. Tarama sınırının sonunu belirler.

8. DOSYA BİRLEŞTİRME DEYİMLERİ

Program içerisinde ikinci bir programla bağlanma ya da iki dosyayı birleştirme gibi durumlarda bu bölümde göreceğimiz deyimlerden yararlanılır.

a. MERGE

Yazılışı:

```
MERGE <" [Sürücü:]Dosya Adı">
```

Bu komut, dış bellekteki bir programı çağırarak, iç bellekteki programla birleştirmek için kullanılır. Bu komutu kullanırken dikkat etmemiz gereken nokta, iki programın satır numaralarının birbirinin devamı olması gerektiğidir.

Birleştirme yapabilmek için iki programı da **ASCII** kodlarıyla kaydetmemiz gerekir. Örnek verecek olursak:

BORDRO programı ile **SICIL** programını bağlayalım. Önce bu iki programın satır numaralarını sıraya koyalım. Diyelim ki **BORDRO** programı **10**'dan başlayıp **1350**'ye kadar gitsin. O zaman **SICIL** programını da **1400**'den başlatalım. Şimdi **BORDRO** programını **ASCII** olarak diskete kayıt edelim. Ardından **SICIL** programını da diskete **ASCII** olarak kayıt edelim. Sonra aşağıdaki şekilde **BORDRO** programını belleğe yükleyelim.

```
MERGE "BORDRO" ↵
```

Sonra **SICIL** programını yine aşağıdaki şekilde belleğe çağıralım.

```
MERGE "SICIL" ↵
```

Şimdi programın listesini alacak olursak, iki programın tek bir program haline geldiğini görürüz. Artık elimizde iki programı içeren tek bir program var demektir. Bunu da yeni bir adla, ileride kullanmak üzere diskete kayıt edebiliriz.

b. CHAIN

Yazılışı:

```
CHAIN <" [Sürücü:]Program Adı">[, [Satır No] [,ALL]]  
COMMON Değişken Adı [,Değişken Adı]...
```

Bir programda iken başka bir programa bağlanmak amacıyla kullanılır. Satır no isteğe bağlı olarak, kaçınıcı satırdan sonra bağlanılacaksa belirtmek için kullanılır. **ALL**

kullanılırsa, kullandığımız programdaki bütün değişkenlerin diğer programa da geçeceğini belirtir. **COMMON** komutu da bu komutla bağlantılı olarak, belli birkaç değişkenin geçmesi istendiği durumlarda kullanılır.

500 CHAIN "ANAMENU" , 720 , ALL

Program çalışması sırasında bu satıra gelince, o programdan çıkarak, diskette kayıtlı bulunan **ANAMENU** adlı programa, **720.** satırdan itibaren bağlanacak ve tüm değişkenleri de beraberinde aktaracaktır.

Eğer tüm değişkenleri değil de, değişkenlerin yalnızca birkaçını aktaracak olsaydık, **ALL** komutunu yazmadan bir üst satıra **COMMON** komutunu geçmesini istediğimiz değişkenlerle birlikte kullanacaktık.

495 COMMON AC , D , ADŞ , SOYŞ

Bu satırı **CHAIN** komutundan bir satır önce yazdığımızı varsayarsak, **AC , D , ADŞ , SOYŞ** değişkenleri geçtiğimiz programa da geçecektir.

9. DİĞER KOMUT, DEYİM ve FONKSİYONLAR

a. RESET

Yazılışı:
RESET

Tüm dosyaların kapatılmasını sağlar. Dosyaların iş bitiminde kapatılması gerekmektedir. Bu şekilde dosyaların bozulmasını önlemiş oluruz.

Programın çıkış satırına konacak olursa program sona ermeden önce tüm açık dosyalar kapatılmış olur.

9990 RESET

10000 END Şeklinde yazılır.

b. MKDIR

Yazılışı:
MKDIR "Dizin Adı"

Basic ortamında dizin (Directory) yaratır.

Örneğin **GUNCE** adında bir dizin yaratalım.

MKDIR "GUNCE" ↵

c. CHDIR

Yazılışı:
CHDIR "Dizin Adı"

Basic ortamında bir dizin içerisine girmek istersek bu komutu kullanırız. **GUNCE** dizinine girelim.

CHDIR "GUNCE" ↵

Yazarak **GUNCE** dizinine gireriz.

CHDIR ".." ↵

Yazarak bulunduğumuz dizinden çıkarız.

d. RMDIR

Yazılışı:
RMDIR "Dosya Adı"

Bu komutla dizin silinir. Az önce açtığımız **GUNCE** dizinini silelim.

RMDIR "GUNCE" ↵

Şeklinde yazarak **GUNCE** dizinini sileriz. Yalnız şunu unutmamalıyız. Sileceğimiz dizinin bir üstünde olmalıyız.

e. DEF FN

Yazılışı:

DEF FN<Ad> [Değişken Adı],...=<Ifade>

Bu deyim fonksiyonu tanımlar. Program içerisinde daha basit ve istediğimiz kadar kullanmayı sağlar. Değişkenleri aldığı farklı değerlerde korur.

Ad	Eşitliğin sağ tarafında tanımlanan fonksiyona verilen addır.
Değişken Adı	Fonksiyon içerisinde yer alan değişkenlerdir.
İfade	Değişkenlerin değerlerinin matematiksel ifadesidir.

f. DEF

Yazılışı:

DEF <Tip> Harf[-Harf] . . .

Değişkenin türünün tamsayı, tek duyarlı sayı ve çift duyarlı sayı ya da alfasayısal dizgi olmasını tanımlar.

DEFINT	Programdaki değişkenleri tam sayı olarak tanımlar.
DEFSNG	Programdaki değişkenleri tek duyarlı sayı olarak tanımlar.
DEFDBL	Programdaki değişkenleri çift duyarlı sayı olarak tanımlar.
DEFSTR	Programdaki değişkenleri alfasayısal dizgi olarak tanımlar.
DEFINT A-F	A ile F arasındaki harflerle başlayan değişkenler tam sayıdır. (A,B,C,D,E,F)
DEFSNG A-D,X-Z	A,B,C,D ve X,Y,Z ile başlayan değişkenler tek duyarlı sayıdır.
DEFDBL A	A ile başlayan değişkenler çift duyarlı sayıdır.
DEFSTR K-P	K,L,M,N,O,P ile başlayan değişkenler alfasayısaldir.

g. ERR ve ERL

Yazılışı:
Y=ERR
Z=ERL

ERR hata durum kodunu belirtir. **ERL** hatanın bulunduğu satır numarasını belirtir.

h. ON ERROR GOTO-RESUME

Yazılışı:
ON ERROR GOTO <Satır Numarası>

Bu deyim, bu şekilde yazılışında programın çalışması sırasında bir hata olması durumunda, Satır Numarası belirtilen hata alt programına gider.

Satır numarası yerine **0** verilmesi durumunda hata mesajı yazılır ve program durur. Hata alt programının son deyim **RESUME <Satır No>** ile biterse, programın akışı verilen satır numarasından devam ederek döner ve hata yinelenir. **RESUME NEXT** deyimini kullanılmışsa, program kaldığı yerden bir kereye özgü olmak üzere devam eder.

```
10 ON ERROR GOTO 70
20 REM BÖLME PROGRAMI
30 INPUT "BÖLÜNEN SAYI." ;BS
40 INPUT "BÖLEN SAYI." ;BLS
50 SONUC=BS/BLS
55 PRINT SONUÇ
60 END
70 PRINT "SIFIRA BÖLME VAR"
80 PRINT "HATA TÜRÜ";ERR 11
90 PRINT "HATA KODU";ERL 50
100 RESUME NEXT
```

Bu programı çalıştırdığımızda, sıfıra bölme hatası yapıldığında, program hata mesajını ve hatanın türü ile kodunu verecektir.

i. FRE

Yazılışı:
FRE (X)
FRE (X\$)

BASIC tarafından bellekte kullanılmayan alanı byte olarak verir. **X** sayısal değeri verildiğinde bellekte kullanılmayan boş alanı verir. **X\$** karakter dizi şeklinde verilirse, boş alanı belirtmeden önce bellek temizlenerek kullanılan alanlar sıkıştırılır.

```
PRINT FRE (0)  
60362
```

```
PRINT FRE (A)  
60362
```

j. PEEK

Yazılışı:
PEEK (n)

n ile belirlenen, bellek konumunda depolanan **1 byte**'lık bilgiyi okuyarak programa aktarır. Bu deyimın tamamlayıcısı **POKE** deyimidir.

```
10 AS=PEEK(2580)  
20 PRINT "AS=";AS      gibi...
```

```
10 DEF SEG=&HB000  
20 PRINT PEEK (0)
```

Ekranın sol üst köşesindeki karakterin, karakter karşılığını yazar. **PEEK (1)** olursa renk numarasını verir. Grafik kartı renkli ise **&HB800** adresi kullanılmalıdır.

k. POKE

Yazılışı:
POKE (n ,m)

Bilgisayarın bellek konumuna, **1 byte**'lık bilgi yazar.

n Bilgisayarın bellek sınırları arasında bir tam sayıdır. Bellekte ayrılan adrestir.

m n ile tanımlanan konuma yazdırılmak istenen karakterin kodudur. Değeri 0-255 arasındadır.

10 POKE 98,219 gibi...

```
10 DEF SEG=&HB000
20 POKE 0,3
```

Ekranın sol üst köşesine bir karakteri yazar. **POKE 1,15** olursa sol üstteki karakter parlar.

PEEK ve **POKE** komutları ile bilgisayarın belleğinde, çeşitli oynama ve değişiklikler yapabilirsiniz.

I. KEY

Yazılışı:

```
KEY <Fonksiyon Tuşu No>,<Alfasayısal İfade>
KEY LIST
KEY ON
KEY OFF
```

Bu deyim, belirli bir fonksiyon tuşuna başka bir görev yüklemek için kullanılır.

Fonksiyon Tuşu No	Değiştirilmek/görev yüklenmek istenen tuş numarası.
Alfasayısal İfade	Fonksiyon tuşuna programlanan komut.
KEY LIST	Tuşların listelenmesini sağlar.
KEY ON	Fonksiyon tuşlarının görüntülenmesini sağlar.
KEY OFF	Alt satırda fonksiyon tuşlarının durumunu gösteren tabloyu silerek, o satırın da kullanıcı tarafından kullanılmasını sağlar.

KEY deyimini aşağıdaki yazılış biçimiyle program içerisinde kullanabiliriz:

```
KEY x,CHR$(y),+CHR$(z)
```

x	Key numaraları yerine 15'den 20'ye kadar numara
y	Tuşun üst kısmındaki karakter (shift) yerine
z	Herhangi bir tuş numarası

m. ON KEY

ON KEY veya **KEY (x) ON** şeklinde kullanılışları aşağıdaki biçimdedir:

x yerine tuş numarası belirtilir.

```
10 CLS
20 KEY (1) ON          ' 1 ve 2 numaralı fonksiyon tuşları açılıyor.
30 KEY (2) ON
40 ON KEY (1) GOSUB 100 ' Fonksiyon tuşları görev yerlerine gönderiliyor.
50 ON KEY (2) GOSUB 200
60 GOTO 10
..
..
100 KEY 1, "FILES"+CHR$(13) ' 1 numaralı Fonksiyon tuşuna FILES görevi
110 RETURN            ' yükleniyor ve dönüş sağlanıyor
..
..
200 KEY 2, "WIDTH 40"+CHR$(13) ' 2 numaralı fonksiyon tuşuna WIDTH 40
210 KEY (2) OFF      ' görevi yükleniyor ve bu görev kapatılıyor.
220 RETURN          ' Geri dönüş sağlanıyor.
```

Programı çalıştırdıktan sonra **F1** tuşuna basarsak, **FILES** görevi üstlenerek dosya rehberini listeleyecektir. **F2** tuşuna basarsak **WIDTH 40** görevini yüklenerek ekranı **40** karakterlik duruma getirecek ve sonra bu tuşun görevi kapanacaktır.

J. DOSYA DÜZENLEMESİ

Buraya kadar yaptığımız programlarda verileri, ya klavyeden (**INPUT**) girdik ya da program içerisinden **LET, READ-DATA** ile atadık.

Bu komutlarla ve bugüne kadar yaptığımız programlarımızda, verileri her defasında yeniden giriyorduk; bilgisayarımızı kapattığımız anda ya da programı bellekten sildikten sonra veriler, yok oluyordu. Yeni göreceğimiz dersimizde, girilen veriler diskette gerektiğinde kullanılmak üzere saklanabilir, sık sık aynı verileri girmekten kurtuluruz; yalnızca değişebilen veriler girilerek zamandan ve emekten kazanırız. Bu şekilde diskette hem daha çok bilgi toplayabiliriz, hem de istediğimizde bu bilgilerden yararlanabiliriz.

Yukarıda belirttiğimiz türden dosyalara **DATA** (bilgi, veri) dosyaları denilir. Bu tür dosyalara erişebilmek için iyi bir dosya organizasyonu gerekmektedir. **DATA** dosyaları üçe ayrılır:

1. Sıralı Erişimli Dosyalar (Sequential Access Files)
2. Doğrudan Erişimli Dosyalar (Random Access Files)
3. Index Erişimli Dosyalar (Index Sequential Access Files)

1. SIRALI ERİŞİMLİ DOSYALAR

a. SIRALI DOSYALARDA KULLANILAN KOMUTLAR

Sıralı dosyalarda, bilgiye erişmek istediğimizde, dosya içerisindeki tüm bilgiler okunur, taranır. Tarama sonunda, erişmek istediğimiz bilgi karşımıza gelir.

Sıralı erişimli dosyalarda aşağıdaki deyimler kullanılır:

OPEN	Dosya açma deyimi.
CLOSE#	Dosya kapama deyimi.
LINE INPUT#	254 karaktere kadar bilgi girme.
PRINT#-WRITE#	Dosyaya bilgi kaydı.
INPUT#	Dosyadan bilgi okuma.
EOF	Dosya sonu kontrol deyimi.
LOF	Dosyadaki byte sayısını verir.
LOC	Bilginin kayıt numarasını verir.

1) DOSYA AÇMA DEYİMİ

OPEN

Yazılışı:

```
OPEN <"Mod">, [#]<Dosya No>, <" [Sürücü:] \dizin Adı \
                               Dosya Adı">[, Kayıt Uzunluğu]
```

veya

```
OPEN <" [Sürücü:] \Dizin Adı \Dosya Adı">, FOR<"Mod"> AS
                               [#]<Dosya No>[, LEN=Kayıt Uzunluğu]
```

OPEN deyimi, dosyadan hangi amaçla olursa olsun (Kayıt, Okuma, Düzeltme) dosyayı açmak için kullanılır. Sıralı erişimli dosyalar üç tipte açılır:

Mod	Amaç
O (Output)	Dosyayı bilgi kaydı, yaratmak amacıyla açmak için.
I (Input)	Dosyayı bilgi okumak amacıyla açmak için.
A (Append)	Dosyayı bilgi eklemek amacıyla açmak için kullanılır.

Not: **O** modu dosya yaratmak amacıyla açılır. Programı ilk çalıştırdığımızda dosya yaratılır. Daha sonra aynı programı çalıştırırken **O** modu ile bir kez daha açarsak daha önce kaydettiğimiz bilgiler silinecektir. Çünkü dosya yeniden yaratılacaktır.

Sürücü: **DATA** verilerinin çalıştığımız sürücüye değil de, başka bir sürücüye kayıt edilmesini istiyorsak sürücüyü belirtiriz. Sürücü belirtilmezse, dosya işlemleri o an çalışılmakta olan sürücüde yürütülecektir.

Sıralı erişimli dosyalarda, kayıt uzunluğunu belirtmek zorunluluğu yoktur.

Dosya Adı: Daha önce verdiğimiz kurallara uyularak verilir. Uzantı verip vermemek isteğe bağlıdır. Bu tip dosyalarda uzantı genelde veri dosyası olduğunu belirtmek için **.DAT**'tir.

Dosya açmaya birkaç örnek yapalım:

OPEN "O", #1, "MUSTERI.DAT"	1 numaralı MUSTERI.DAT adlı dosyayı kayıt, yaratmak amacıyla açtık.
OPEN "I", #2, "STOK.DAT"	2 numaralı STOK.DAT adlı dosyayı bilgi almak, okumak amacıyla açtık.
OPEN "A", #3, "SENET"	3 numaralı SENET adlı dosyayı bilgi eklemek amacıyla açtık.

2) DOSYA KAPAMA DEYİMİ

CLOSE

Yazılışı:

CLOSE [#]<Dosya No>, . . .

OPEN ile açılan bir dosyayı kapatmak için kullanılır. İş bitiminde açık olan dosya mutlaka kapatılmalıdır; çalıştığımız dosya ile işlem bitiyorsa o dosyayı kapatmadan, programı kesmememiz gerekir. Bunu yapmadığımız takdirde **DATA** dosyası bozulabilir.

Bu olayı şuna benzetebiliriz: Bir kutu ya da bir dolaptan herhangi bir şey almak için nasıl kapağını açıyor ve aldıktan sonra tekrar kapatıyorsak burada da aynısını yapıyoruz.

Dosya No verilirse yalnızca numarası verilen dosya kapatılır. Verilmezse açık olan bütün dosyalar kapatılır. **NEW, LOAD, RUN,** ve **END** komutlarından sonra açık olan bütün dosyalar kapatılır.

CLOSE #1 Yalnızca 1 numaralı dosya kapanır.
CLOSE #1,#2,#3 1, 2 ve 3 numaralı dosyalar kapanır.
CLOSE Açık olan bütün dosyalar kapatılır.

3) BİLGİ KAYIT DEYİMİ

PRINT

Yazılışı:

PRINT #<Dosya No>, <Değişkenler>, . . .

O kayıt amacıyla ya da **A** bilgi ekleme amacıyla açılan dosyaya kayıt yapar. **PRINT** komutu ile ekrana yazdırdığımız gibi, bu komutla da bilgileri diskete yazdırırız, kayıt ederiz.

İfadeler ve değişkenler arasında (;) noktalı virgül varsa diskete de sıkışık kayıt eder, (,) virgül varsa aralıklı olarak kayıt eder. Bu işaretler ekran kullanımında olduğu gibi diskette de görülmezler.

PRINT #1,AD\$;SN;GR\$

Bir numaralı sıra erişimli dosyaya **AD\$, SN** ve **GR\$** değişkenlerindeki bilgiler diskete aşağıdaki şekilde kayıt edilir.

ISMAIL TANIŞ	2344	ÖĞRETMEN
--------------	------	----------

4) BİLGİ KAYIT DEYİMİ

WRITE

Yazılışı:

WRITE #<Dosya No>, <Değişkenler>...

O kayıt (yaratma) veya **A** ekleme amacıyla açılmış dosyaya ifadeleri "... " içerisinde kayıt eder.

WRITE #1,AD\$;SN;GR\$ şeklinde yazılır.

Bu deyim ile yapılan kayıtlar da:

"ISMAIL TANIŞ";"2344";"ÖĞRETMEN"

şeklinde diskette yer alır.

5) BİLGİ OKUMA DEYİMİ

INPUT

Yazılışı:

INPUT #<Dosya No>, <Değişkenler>, ...

I tipinde, bilgi okumak amacıyla açılmış dosya için diskette kayıtlı bilgileri değişkenlere yükleyerek belleğe aktarır. **INPUT** deyiminde olduğu gibi soru işareti görüntülenmez.

INPUT #1,AD\$;SR;GR\$ Şeklinde yazılarak, diskette kayıtlı bilgiler belleğe değişkenler aracılığıyla aktarılır.

6) DOSYA SONU KONTROL FONKSİYONU

EOF

Yazılışı:

EOF (<Dosya No>)

Bu fonksiyon dosyanın sonuna erişilip erişilmediğini kontrol eder. **EOF** fonksiyonunun değeri normal zamanda **0**'dir. Dosya sonunda **-1** değerini alır.

IF EOF (1) THEN 200 1 numaralı dosyanın sonuna gelince **200.** satıra gider.

7) KAYIT UZUNLUĞU BELİRTME FONKSİYONU LOF

Yazılışı:
LOF (<Dosya No>)

Dosyadaki kayıtların **byte** olarak uzunluklarını verir.

Kayıt Sayısı=LOF(1)/Kayıt Uzunluğu formülü ile çalışır.

FOR I% = 1 TO LOF(1)/86

8) SON KAYIT BELİRTME FONKSİYONU LOC

Yazılışı:
LOC (<Dosya No>)

Açık olan dosyadaki son kayıdın numarasını verir.

IF LOC(1)=25 THEN 500 gibi.

9) KARAKTER DİZİSİNİ KAYIT DEYİMİ LINE INPUT

Yazılışı:
LINE INPUT #<Dosya No>,<String Değişken>

255 karaktere kadar olan karakter dizisini, okuyarak tek string göze yerleştirir. Daha önce **INPUT** deyimiyile bazı karakterleri giremiyorduk, Örneğin , (virgül) gibi. **LINE INPUT** ile kaydettiğimiz bilgileri **LINE INPUT#** ile belleğe yükleriz. Bu deyimle (,) dahil benzer karakterleri girebiliriz. Tek bir satırda birkaç cümleciği girmek istediğimiz zaman kullanılır.

Dosya **O** modunda ya da **A** modunda açılır:

LINE INPUT "ADRESİ GIR (MAHALLE,CADDE,SOKAK, NO) ";ADR\$

Bilgiler aralara virgül de konarak girilir, diskete kayıt edilir.

Dosya **I** modunda açılır:

LINE INPUT #1,ADR\$

Bilgiler disketten okunur.

b. SIRALI ERİŞİMLİ DOSYA UYGULAMALARI

Sıralı erişimli dosyaya örnek program yapalım:

SIRALI ERİŞİMLİ DOSYALARDA KAYIT AMACIYLA DOSYA AÇMA

```
10 REM BU PROGRAM "KITAP.DAT" ADLI
20 REM DOSYAYI YARATMAK AMACIYLA AÇAR
30 REM BU PROGRAMI "KAYITS" ADI İLE DISKETE KAYIT EDİN
40 OPEN "O",#1,"KITAP.DAT"
50 CLS:PRINT "ÇIKIŞ İÇİN SIRA NO'YA (0) GIRINIZ"
60 INPUT "SIRA NO";NO!
70 IF NO!=0 THEN CLOSE #1:END
80 INPUT "KITABIN ADI";KAŞ
90 INPUT "KITABIN YAZARI";KYŞ
100 INPUT "KITABIN TÜRÜ";KTŞ
110 INPUT "KITABIN FİYATI";KF!
120 PRINT :PRINT
130 INPUT "GİRDİĞİNİZ BİLGİLER DOĞRU MU? (E/H)";YŞ
140 IF YŞ="H" OR YŞ="h" THEN PRINT "YENİDEN GIRINIZ":GOTO 50
150 WRITE #1,NO!,KAŞ,KYŞ,KTŞ,KF!
160 GOTO 50
```

PROGRAMIN SATIR SATIR AÇIKLAMASI:

- 40. satır:** KITAP.DAT adlı dosya kayıt ve yaratma amacıyla açılıyor.
60. satır: Sıra no girişi yaptırılıyor.
70. satır: Kontrol satırıdır. Çıkış için kontrol ediliyor. Giriş sıfır ise dosya kapatılıp, program sona erdiriliyor.
80-110 satırlar: Klavyeden giriş yapılıyor.
120. satır: İki satır boş bırakılıyor.
130. satır: Girilen bilgilerin doğruluğu onaylanıyor.
140. satır: Bilgi girişi kontrolü yapılıyor. Yanlış giriş varsa yeni giriş geri gönderiliyor.
150. satır: Sıralı erişimli dosyaya girilen bilgiler, değişkenler aracılığı ile kayıt ediliyor.
160. satır: Yeni giriş için giriş satırına gönderiliyor.

Bu yaptığımız programı, Sıralı Erişimli Dosya olduğunu anlamak için sonuna S harfini ekleyerek, **KAYITS** adı ile diskete kayıt edelim. Çalıştırıp birkaç bilgi girelim.

Not: Kayıt programını ilk kayıta kullanmalıyız. Sonraki kayıtlarda ekleme programını kullanmamız gerekir. Eğer bilgi girişinden sonra kayıt programını yeniden çalıştıracak olursak, yeni bir dizi açılacağından önceki bilgiler silinecektir. Kısaca buradaki kayıt programı bir yaratma programıdır.

SIRALI ERİŞİMLİ DOSYALARDA BİLGİYE ERİŞİM AMACIYLA DOSYA AÇMA

```
10 CLS
20 ' BU PROGRAM SIRALI ERİŞİMLİ DOSYADA
30 ' KAYITLI BİLGİLERE ERİŞİMİ SAĞLAR
40 ' BU PROGRAMI "ERISIMS" ADI İLE DISKETE KAYIT EDİNİZ
50 PRINT "1- KİTABIN YAZARINA GÖRE ERİŞİM"
60 PRINT "2- KİTABIN TÜRÜNE GÖRE ERİŞİM"
70 PRINT "3- PROGRAMDAN ÇIKIŞ"
80 PRINT :PRINT
90 INPUT "SEÇİMİNİZİ RAKAM OLARAK GIRINIZ";S
100 IF S<1 OR S>3 THEN BEEP:BEEP:GOTO 90
110 OPEN "I",#1,"KITAP.DAT"
120 ON S GOTO 130,250,370
130 PRINT :PRINT :PRINT
140 INPUT "KİTABIN YAZARINI GIRINIZ";KAYŞ
150 GOSUB 380
160 WHILE NOT EOF(1)
170 INPUT #1,NO!,KAŞ,KYŞ,KTŞ,KF!
180 IF KYŞ=KAYŞ THEN PRINT NO!,KAŞ,KYŞ,KTŞ,KF!
190 WEND
200 CLOSE #1
210 PRINT :PRINT :PRINT
220 PRINT "DEVAM İÇİN BİR TUŞA BASINIZ"
230 IF INKEY$="" THEN 230
240 GOTO 10
250 PRINT :PRINT :PRINT
260 INPUT "KİTABIN TÜRÜNÜ GIRINIZ";KATŞ
270 GOSUB 380
280 WHILE NOT EOF(1)
290 INPUT #1,NO!,KAŞ,KYŞ,KTŞ,KF!
300 IF KTŞ=KATŞ THEN PRINT NO!,KAŞ,KYŞ,KTŞ,KF!
310 WEND
320 CLOSE #1
330 PRINT :PRINT :PRINT
340 PRINT "DEVAM İÇİN BİR TUŞA BASINIZ"
350 IF INKEY$="" THEN 350
360 GOTO 10
370 CLOSE #1:END
380 PRINT "SIRA NO","KİTABIN ADI","YAZARI","TÜRÜ","FIYATI"
390 PRINT "=====","=====","=====","=====","====="
400 RETURN
```

PROGRAMIN SATIR SATIR AÇIKLAMASI:

- 50-100. satırlar:** Ekran menüsü ve seçenekler oluşturuluyor.
110. satır: Dosya I modunda, kayıt okumak amacıyla açılıyor.
120. satır: Alt programlara gönderme yapılıyor.
140. satır: Aranılan kayıt girişi yaptırılıyor.
150. satır: Başlık çizimi için alt programa gönderiliyor.
160. satır: WHILE-WEND döngüsü başlatılarak tüm dosya içerisinde döngü sağlanıyor. Döngü dosya sonuna kadar sürüyor.
170. satır: Dosya içerisindeki veriler okutuluyor.
180. satır: Girilen bilgi ile dosyadaki bilgiler karşılaştırılıyor. Koşul yerine geliyorsa ekrana yazdırılıyor.
190. satır: Döngü kapatılıyor.
200. satır: Dosya kapatılıyor.
210. 230. satırlar: Bilgilerin görülmesi için ekran bekletiliyor. Herhangi bir tuşa basınca kaldığı yerden devam ediyor.
240. satır: Ekran menüsüne dönülüyor.
250-360. satırlar: Başka arama yöntemi için alt program. Yöntem aynı.
370. satır: Dosya kapatılması ve çıkış.
380-400. satırlar: Başlık alt programı

(Bu programı diskete ERISIMS adı ile kayıt ediniz.)

SIRALI ERİŞİMLİ DOSYALARDA YENİ BİLGİ EKLEMEK İÇİN DOSYA AÇMA

```
10 CLS
20 ' BU PROGRAM "KITAP.DAT" ADLI DOSYAYA
30 ' BILGI EKLER
40 ' BU PROGRAMI "EKLEMES" ADI ILE DISKETE KAYIT EDINIZ.
50 OPEN "A",#1,"KITAP.DAT"
60 CLS
70 INPUT "SIRA NO";NO!
80 INPUT "KITABIN ADI";KA$
90 INPUT "KITABIN YAZARI";KY$
100 INPUT "KITABIN TÜRÜ";KT$
110 INPUT "KITABIN FİYATI";KF!
120 INPUT "GİRDİĞİNİZ BİLGİLER DOĞRU MU? (E/H) ...>";YN$
130 IF YN$="H" OR YN$="h" THEN 60
140 WRITE #1,NO!;KA$;KY$;KT$;KF!
150 INPUT "BAŞKA GİRİŞ VAR MI (E/H) ... ";Y$
160 IF Y$="E" OR Y$="e" THEN 60
170 CLOSE #1:END
```

PROGRAMIN SATIR SATIR AÇIKLAMASI:

50. satır:	Dosya A modunda bilgi eklemek için açılıyor.
70. satır:	Sıra numarası girişi.
80-110. satırlar:	Bilgi girişi için.
120-130. satırlar:	Bilgilerin doğruluğu onaylanıyor.
140. satır:	Yeni bilgiler diskete kayıt ediliyor.
150-160. satırlar:	Başka giriş olup olmadığı soruluyor.
170. satır:	Dosya kapatılıyor ve programdan çıkılıyor.

(Bu programı diskete **EKLEMES** adı ile kayıt ediniz.)

Bu üç programın çalışması birbiri ile bağıntılıdır. **KAYITS** adlı ilk programla kayıt girilir. **ERISIMS** adlı programla dosyadaki bilgiler görüntülenir. **EKLEMES** adlı programla, **KITAP.DAT** adlı dosyaya yeni bilgiler eklenir.

Şimdi, **KAYITS** adlı programı belleğe yükleyelim ve birkaç adet bilgi girelim. Bilgi kaydı bitince çıkmak için sıra no'ya **0** giriniz. Sonra **ERISIMS** adlı programı belleğe yükleyelim ve erişimi hangi seçeneğe göre yaparsak seçim numarasını girerek, istediğimiz bilgilere erişelim. Ekleme yapma gereğini duyarsak, **EKLEMES** adlı programı belleğe yükleyelim ve yeni girişlerimizi yapalım.

Sıralı erişimli dosyaya bilgi eklemede ikinci bir yöntem de, değişiklik yapmak için de kullanılan Ana Dosyayı okutup, kayıt etiketten sonra, yeni bir dosya açarak bu dosyada işlemleri yapmaktır.

```
10 CLS
20 ' SIRALI DOSYAYA BILGI EKLEME
30 ' "EKLEMES2" ADI ILE DISKETE KAYIT EDINIZ
40 OPEN "I",#1,"KITAP.DAT"
50 OPEN "O",#2,"YENIDOSY"
60 INPUT #1,NO!,KA$,KY$,KT$,KF!
70 WRITE #2,NO!;KA$;KY$;KT$;KF!
80 IF EOF(1) THEN 100
90 GOTO 60
100 CLOSE #1
110 ' YENI BILGILERIN EKLENMESI
120 CLS
130 INPUT "SIRA NO";NO!
140 IF NO!=0 THEN 210
150 INPUT "KITABIN ADI";KA$
160 INPUT "KITABIN YAZARI";KY$
170 INPUT "KITABIN TURU";KT$
```

```
180 INPUT "KITABIN FİYATI";KF!  
190 WRITE #2,NO!,KAŞ,KYŞ,KTŞ,KF!  
200 GOTO 120  
210 CLOSE #2  
220 KILL "KITAP.DAT"  
230 NAME "YENIDOSY" AS "KITAP.DAT"  
240 END
```

PROGRAMIN SATIR SATIR AÇIKLAMASI:

40. satır: Ana dosya **I** modunda açılıyor.
50. satır: Geçici dosya **O** modunda açılıyor.
60. satır: Ana dosyada kayıtlı bilgiler okutuluyor.
70. satır: Ana dosyadaki bilgiler geçici dosyaya kayıt ediliyor.
80. satır: Dosya sonu kontrol ediliyor. Son ise ana dosyanın kapatılacağı satıra gönderiliyor.
90. satır: Okuma işleminin sürdürülmesi için okuma satırına gönderiliyor.
100. satır: Ana dosya kapatılıyor.
(110 ve 210. satırlardaki işlemlerde kayıt işlemleri geçici dosyada sürdürülüyor.)
130. satır: Sıra no girişi.
140. satır: Kontrol satırı.
150-180. satırlar: Bilgi girişi.
190. satır: Girilen bilgiler, geçici dosyaya kayıt ediliyor.
200. satır: Yeni giriş için, giriş satırına gönderiliyor.
210. satır: Geçici dosya kapatılıyor.
220. satır: Ana dosya disketten siliniyor.
230. satır: Geçici dosyanın adı ana dosya adı ile değiştiriliyor.
240. satır: Program sona eriyor.

Sıralı dosyada bilgi değişikliği yapacaksa aşağıdaki yöntemi uyguluyoruz:

```
10 CLS  
20 ' SIRALI DOSYADA BİLGİ DEĞİŞİKLİĞİ  
30 ' "DEĞİŞİMS" ADI İLE DISKETE KAYIT EDİNİZ  
40 INPUT "DEĞİŞTİRİLECEK KAYDIN NUMARASINI GIRIN";N!  
50 OPEN "I",#1,"KITAP.DAT"  
60 OPEN "O",#2,"YENIDOSY"  
70 S=0  
80 IF EOF(1) THEN 250  
90 INPUT #1,NO!,KAŞ,KYŞ,KTŞ,KF!  
100 IF NO!<>N! OR S<>0 THEN WRITE #2,NO!,KAŞ,KYŞ,KTŞ,KF! :
```

```
GOTO 80
110 ' BILGININ DEĞİŞTİRİLMESİ
120 PRINT "KİTABIN ADI.....: ";KAŞ
130 PRINT "KİTABIN YAZARI.....: ";KYŞ
140 PRINT "KİTABIN TÜRÜ.....: ";KTŞ
150 PRINT "KİTABIN FİYATI.....: ";KF!
160 INPUT "DEĞİŞTİRİLECEK KAYIT BU MU? (E/H) ....>";YNTŞ
170 IF YNTŞ="H" OR YNTŞ="h" THEN 240
180 PRINT :PRINT :PRINT :PRINT
190 INPUT "SIRA NO";NO!
200 INPUT "KİTABIN ADI";KAŞ
210 INPUT "KİTABIN YAZARI";KYŞ
220 INPUT "KİTABIN TÜRÜ";KTŞ
230 INPUT "KİTABIN FİYATI";KF!
240 S=1:WRITE #2,NO!,KAŞ,KYŞ,KTŞ,KF!:GOTO 80
250 IF S=0 THEN PRINT:PRINT :PRINT "BÖYLE BİR KAYIT YOK"
260 CLOSE
270 KILL "KİTAP.DAT"
280 NAME "YENIDOSY" AS "KİTAP.DAT"
290 END
```

PROGRAMIN SATIR SATIR AÇIKLAMASI:

- 40. satır:** Değiştirilecek kayıt numarasının girişi.
- 50. satır:** Ana dosya **I** modunda kayıt okumak amacıyla açılıyor.
- 60. satır:** Geçici dosya, **O** modunda kayıt amacıyla açılıyor.
- 70. satır:** **S** gözü kayıt kontrolü için **Sıfır** olarak açılıyor.
- 80. satır:** Dosya sonu kontrol satırı. Kayıt bulunamazsa belirtme satırına gönderiliyor.
- 90. satır:** Ana dosyadaki bilgiler disketten okunuyor.
- 100. satır:** Kontrol ve kayıt satırı. Sorulan kayıt ile disketteki kayıtlar karşılaştırılıyor ve **S** gözü sıfıra eşit değil ve kayıtlar tutmuyorsa geçici dosyaya aktarılıyor. Dosya sonu kontrol satırına gönderiliyor.
- 120-150. satırlar:** Değiştirilecek bilgi ekranda görüntüleniyor.
- 160-170. satırlar:** Doğruluk onaylatılıyor.
- 190-230. satırlar:** Değişiklikler giriliyor.
- 240. satır:** Yeni bilgilerin girişinde **S** gözü **1** değerini alıyor. Böylece kayıtlar tutsa dahi geçici dosyaya kayıt yapılıyor. Dosyanın sonuna kadar taranması için, dosya sonu kontrol satırına gönderiliyor.
- 250. satır:** **S** gözü **Sıfır** ise böyle bir kayıt bulunmadığından, mesaj verdiriliyor.

260. satır:	Dosyalar kapatılıyor.
270. satır:	Ana dosya siliniyor.
280. satır:	Geçici dosyanın adı ana dosya ile değiştiriliyor.
290. satır:	Program sona erdiriliyor.

İsterseniz, bu programları tek program haline getirebilirsiniz. **ON...GOTO** ile birleştirip veya **CHAIN** ile bağlayıp, daha kullanışlı bir program elde etmeyi deneyin.

2. DOĞRUDAN (RANDOM) ERİŞİMLİ DOSYALAR

Sıralı erişimli dosyalarda, bilgiye erişmek için tüm dosyayı taraması gerekiyordu, bu da zaman kaybına yol açıyordu. Doğrudan erişimli dosyalarda bu sorunu da çözümlüyoruz, ulaşmak istediğimiz bilginin kayıt numarasını girerek anında ulaşımı sağlamış oluruz.

RANDOM (Rastgele) dosyalarda, dosya açma modu yalnızca **R**'dir. Dosyayı hangi amaçla açacak olursak olalım **R** modunda açarız. Doğrudan erişimli dosyalar **Buffer** kullanırlar. Bu dosyalarda kayıt uzunluğu belirlenir. Doğrudan erişimli dosyalarda, tüm bilgiler karakter bilgiye çevrilerek kayıt edilir. Okumak gerektiğinde karakter bilgi olarak kaydettiğimiz sayısal bilgiler, tekrar sayısal bilgiye çevrilirler.

Doğrudan erişimli dosyalarda **BASIC** ortamında normal açılışta kayıt uzunluğu **128** karakterdir. Dosyamızın kayıt uzunluğu **128** karakterden çok ise **/S** anahtarını kullanarak **BASIC**'i açarız. Bu türden açış için, sistemden **BASIC**'e geçerken kayıt uzunluğunu vererek geçmeliyiz. **150** karakterlik bir kayıt alanı belirlemiş isek, **GW BASIC /S:150** yazarak geçmeliyiz. Normal olarak bir programda en çok **3** dosya açılabilir. Eğer dosya sayısı bu barajı aşılırsa **/F** anahtarını kullanırız. **6** dosyalı bir programımız varsa, **GW BASIC /F:6** şeklinde geçiş yapmalıyız. İki anahtarı birlikte kullanabiliriz.

a. DOĞRUDAN ERİŞİMLİ DOSYALARDA KULLANILAN

KOMUTLAR

OPEN	Dosya açma deyimi. R modunda açılır. Kayıt uzunluğu belirtilir.
FIELD	Kullanılan Buffer, değişkenlere pay edilir.
CLOSE #	Dosya kapatma deyimi.
PUT #	Girilen bilgiler diskete kayıt edilir.
GET #	Disketteki kayıtlı bilgiler, okunur.
LSET-RSET	Bilgiler sola-sağa dayalı olarak kayıt edilir.
MKIS-MKSS-MKDS-STR\$	Sayısal bilgiler, karakter bilgiye dönüştürülür.
CVI-CVS-CVD-VAL	Diskette kayıtlı sayısal bilgiler, karakter bilgidan tekrar sayısal bilgiye çevrilir.
LOC	Kayıt numarasını belirtir.
LOF	Dosyadaki byte sayısını verir.

Sıralı erişimli dosyalarda gördüğümüz bazı deyimler, doğrudan erişimli dosyalarda da kullanılmaktadır. Sıralı erişimli dosyalarda verdiğimiz deyimleri burada bir kez daha yinelemeyeceğiz. Yazılışları ve kullanılış amaçları aynıdır.

1) DOSYA AÇMA DEYİMİ

OPEN

Yazılışı:

```
OPEN "<Mod>",<Dosya No>,<"[Sürücü:]Dosya Adı">,<Kayıt Uzunluğu>
```

Mod sürekli **R**'dir. Dosyayı hangi amaçla açacak olursak olalım **R** modunda açarız. Kayıt için, bilgi almak için ve eklemek için **R** modunda açılır.

Kayıt Uzunluğu belirtilmezse yorumlayıcı **128 byte**'lık alan ayıracaktır. Belirtirsek, verdiğimiz alan kadar yer ayıracaktır. Burada belirttiğimiz kayıt uzunluğu **FIELD** deyimiyile değişkenlere bölüştürülecektir.

```
OPEN "R",#1,"STOK.DAT",28
```

1 nolu **STOK.DAT** adlı dosya, doğrudan erişimli olarak **28 byte**'lık kayıt uzunluğunda açıldı.

2) BUFFER'İ DEĞİŞKENLERE PAYLAŞTIRMA

FIELD

Yazılışı:

```
FIELD #<Dosya No>,<Değişkene Düşen Alan Uzunluğu> AS<Dizi Değişken>,...
```

OPEN ile dosyayı açtıktan sonra, **FIELD** deyimiyile alan uzunluğu değişkenlere bölüştürülür.

```
FIELD #1,15 AS MA$,6 AS SMK$,7 AS F$
```

1 nolu dosyadaki kayıt uzunluğu değişkenlere bölüştürüldü. Burada dikkat etmemiz gereken nokta, **OPEN** deyiminde verilen kayıt uzunluğu ile **FIELD** deyiminde paylaşılan uzunlukların toplamı birbirine eşit olmalıdır.

MA\$=MALIN ADI	15 Karakterlik alan ayrıldı.
SMK\$=MALIN MİKTARI	6 Karakterlik alan ayrıldı.
F\$=FİYATI	7 karakterlik alan ayrıldı.
Toplam	28 karakterlik alan ayrıldı.

Burada belirttiğimiz alan uzunluğunu geçen bir bilgi girersek, uzunluktan fazlası dikkate alınmayacaktır. **MALIN ADI** için **18** karakterlik bir veri girdiysek, son **3** karakter gözönüne alınmayacaktır.

Dikkat ettiyseniz **MALIN MİKTARI** ve **FİYATI** sayısal olduğu halde, **\$** işareti koyduk. Doğrudan erişimli dosyalarda, bütün bilgilerin karakter bilgiye çevrilerek kayıt edildiğini söylemiştik. Bu nedenle sayısal bilgileri de karakter bilgiye çevirmemiz gerekmektedir. Burada daha başlarken, değişken adı tanımlamasında buna dikkat ediyoruz.

Bu dosyalarda bilgiler, daha önceden belirtilen alan uzunlukları ile kayıt edileceğinden, dezavantajı daha kısa bilgi girildiğinde verilen kayıt uzunluğundan kalan kısmın boş yere işgal edilmesidir.

B E L L E K		
MALIN ADI	STOK.MİKTARI	FİYATI
DETERJAN	250	85000
15 Karakter	6 Karakter	7 Karakter

Yukarıda bir alanın belleğe kaydını görüyoruz. Bu anlatım bize, doğrudan erişimli dosya kayıtlarının diskette nasıl yer aldığını gösteriyor.

3) KAYIT DEYİMİ

PUT

Yazılışı:

PUT #<Dosya No>[,<Kayıt No>]

Bu deyim, **FIELD** deyimiyile daha önceden, tampon belleğe aktarılmış olan, **LSET** ya da **RSET** deyimiyile yerleştirilen bilgileri diskete kaydetmek için kullanılır.

PUT #1,K% 1 numaralı dosyada, **K%** ile belirtilen kayıt no'daki bilgiyi diskete kayıt eder.

PUT #1,22 1 numaralı dosyaya **22.** sırada kayıt yapar.

4) BİLGİ OKUMA DEYİMİ

GET

Yazılışı:

GET #<Dosya No>,<Kayıt No>

Bu deyim, daha önceden diskete kaydedilmiş bilgileri okumak için kullanılır.

GET #1,18 1 numaralı dosyadaki **18.** kayıda erişmek için.

ya da

GET #1,K% 1 numaralı dosyadaki **K%** ile belirtilen kayıda erişmek için.

5) SAYILARI DÜZENLEME FONKSİYONLARI

MKI\$, MKS\$, MKD\$, STR\$

Yazılışı:

MKI\$ (<Tamsayı İfade>)
MKS\$ (<Tek Duyarlıklılı İfade>)
MKD\$ (<Çift Duyarlıklılı İfade>)
STR\$ (<Rakama Uygun Çevirme>)

Daha önce, doğrudan erişimli dosyalarda bütün bilgilerin karakter bilgi olarak kaydedildiğini söylemiştik. Bu fonksiyonlarla, sayısal değerler karakter bilgi haline dönüştürülür ve diskete kayıt edilir. **LSET** ya da **RSET** ile birlikte kullanılır.

6) BİLGİLERİ SOLA-SAĞA DAYALI KAYIT

LSET-RSET

LSET diskete kaydedilen bilgileri sola dayalı olarak kaydeder. **RSET** ise bilgileri sağa dayalı olarak kaydeder.

LSET SM\$=MKS\$ (ADT!) **INPUT** girişinde **ADT!** ile tanımlanan, **FIELD** satırında **SM\$** ile tanımlanan, **MALIN MİKTARI**'nı karakter bilgiye çevirir. Sola dayalı olarak kaydeder.

RSET F\$=MKD\$ (FYT#) **FYT#** sayısal değişkenini, **F\$** karakter bilgiye çevirir. (**FİYAT** tanımlanmıştı) Sağa dayalı olarak kaydeder.

7) KARAKTER BİLGİYİ SAYISAL BİLGİYE DÖNÜŞTÜRME

CVI, CVS, CVD

Yazılışı:

A%=CVI (<2 Byte Uzunluğunda String Değişken>)
A!=CVS (<4 Byte Uzunluğunda String Değişken>)
A#=CVD (<8 Byte Uzunluğunda String Değişken>)
A=VAL (<Sayının Kapladığı Alana Göre>)

Daha önce, karakter bilgi haline dönüştürülen sayısal değerleri, tekrar sayısala dönüştürür.

ADT!=CVS (SM\$) Karakter bilgi halindeki **SM\$** stringini, **ADT!** sayısına çevirir.

FYT#=CVD (F\$) Karakter bilgi halindeki **F\$** stringini, **FYT#** sayısına çevirir.

8) DOSYA KAPATMA DEYİMİ

CLOSE

Program sonunda dosyayı kapatırız.

CLOSE #1 1 numaralı dosya kapanır.

b. DOĞRUDAN ERİŞİMLİ DOSYA UYGULAMALARI

Doğrudan erişimli dosyalara kayıt amacıyla bir program aşağıdaki biçimde yapılır:

```
10 REM BU PROGRAM DOĞRUDAN ERİŞİMLİ
20 REM OLARAK KAYIT AMACIYLA DOSYA AÇAR
30 REM "KAYITR" ADI ALTINDA DISKETE KAYDEDİNİZ
40 CLS
50 OPEN "R",#1,"STOK.DAT",29
60 FIELD #1,12 AS MAŞ,8 AS SMŞ,9 AS SFŞ
70 PRINT :PRINT
80 INPUT "KOD NO";KN%
90 PRINT :PRINT :PRINT
100 IF KN%=0 THEN CLOSE #1:END
110 INPUT "MALIN ADI";MADŞ
120 INPUT "STOK MIKTARI";SMK#
130 INPUT "MALIN FİYATI";MFY!
140 LSET MAŞ=MADŞ
150 LSET SMŞ=MKDŞ(SMK#)
160 LSET SFŞ=MKSŞ(MFY!)
170 PUT #1,KN%
180 GOTO 70
```

PROGRAMIN SATIR SATIR AÇIKLAMASI:

50. satır: Dosya **R** modunda, **29 byte** kayıt uzunluğunda açılıyor.
60. satır: Kayıt uzunluğu değişkenlere pay ediliyor.
80. satır: Kod numarası girişi yapılıyor.
100. satır: Kontrol satırı. Çıkış durumunda dosya kapatılıp, program sona erdiriliyor.
110-130. satırlar: Bilgi girişi.
140-160. satırlar: Girilen bilgiler sola dayalı olarak, karakter bilgiye dönüştürülüyor.
170. satır: Bilgiler diskete verilen kod numarasına göre kayıt ediliyor.
180. satır: Yeni bilgi girişi için, giriş satırına gönderiliyor.

Doğrudan erişimli dosyalardan veri okumak amacıyla bir program aşağıdaki biçimde yapılır:

```
10 REM BU PROGRAM DOĞRUDAN ERIŞİMLİ
20 REM OLARAK ULAŞMAK AMACIYLA DOSYA AÇAR
30 REM "ULASIMR" ADI ALTINDA DISKETE KAYDEDİNİZ
40 CLS
50 OPEN "R",#1,"STOK.DAT",29
60 FIELD #1,12 AS MA$,8 AS SM$,9 AS SF$
70 INPUT "KOD NO";KN%
80 IF KN%=0 THEN CLOSE #1:END
90 GET #1,KN%
100 MAD$=MA$
110 SMK#=CVD(SM$)
120 MFY!=CVS(SF$)
130 PRINT "MALIN ADI",MAD$
140 PRINT "STOK MIKTARI",SMK#
150 PRINT "MALIN FİYATI",MFY!
160 GOTO 70
```

PROGRAMIN SATIR SATIR AÇIKLAMASI:

- | | |
|---------------------------|--|
| 50. satır: | Dosya R modunda ve aynı kayıt uzunluğunda açılıyor. |
| 60. satır: | Kayıt alanı değişkenlere pay ediliyor. |
| 70. satır: | Giriş, sorma satırı. |
| 80. satır: | Kontrol satırı. Çıkış durumunda dosya kapatılıyor ve program sona erdiriliyor. |
| 90. satır: | Diskette kayıtlı bilgiler okunuyor. |
| 100-120. satırlar: | Diskette kayıtlı karakter bilgiler tekrar sayısala çevriliyor. |
| 130-150. satırlar: | Bulunan bilgi ekrana görüntüleniyor. |
| 160. satır: | Yeni giriş için, giriş satırına gönderiliyor. |

Doğrudan erişimli olarak bir Müşteri Takip programını aşağıda bulacaksınız:

```
10 ' *****
20 ' *****   EKRAMA BAŞLIK ÇIZDIRILMESI   *****
30 ' *****
40 CLS
50 KEY OFF
60 CLS
70 LOCATE 4,10:PRINT STRING$(60,219)
80 FOR I=4 TO 23
90 SOUND 1111,.1
100 LOCATE I,10:PRINT CHR$(219)
110 LOCATE I,70:PRINT CHR$(219)
120 NEXT I
130 LOCATE 23,10:PRINT STRING$(60,219)
140 COLOR 0,15:LOCATE 6,21:PRINT "ÖZGÜR OFSET MÜŞTERİ TAKIP
PROGRAMI":COLOR 7,0
150 COLOR 15,3:LOCATE 19,23:PRINT " BU PROGRAM MART-1990
TARİHİNDE "
160 LOCATE 20,35:COLOR 31,0:PRINT "GÜVEN TANIŞ":COLOR 7,0
170 COLOR 15,3:LOCATE 21,25:PRINT " TARAFINDAN HAZIRLANMIŞTIR
":COLOR 7,0
180 LOCATE 10,25:PRINT "1. MÜŞTERİ İLK KAYIT GİRİŞİ"
190 LOCATE 11,25:PRINT "2. MÜŞTERİ LİSTESİ ALMA"
200 LOCATE 12,25:PRINT "3. MÜŞTERİ KAYITLARINDA DEĞİŞİKLİK"
210 LOCATE 13,25:PRINT "4. PARA İLE İLGİLİ İŞLEMLER"
220 LOCATE 14,25:PRINT "5. PROGRAMDAN ÇIKIŞ"
230 COLOR 0,7:LOCATE 16,35:INPUT " SEÇİMİNİZ...>";S$:COLOR
7,0
240 IF S$="1" THEN 330
250 IF S$="2" THEN 720
260 IF S$="3" THEN 1790
270 IF S$="4" THEN 2290
280 IF S$="5" THEN 2790
290 IF S$("<>") THEN BEEP:GOTO 230
300 FOR I=23 TO 4 STEP -1
310 LOCATE I,1:PRINT STRING$(80," ")
320 NEXT I
330 ' *****
340 ' *****   KAYIT GİRİŞİ BAŞLIĞI
350 ' *****
360 CLS
370 LOCATE 4,10:PRINT CHR$(201):LOCATE 4,70:PRINT CHR$(187)
380 LOCATE 4,11:PRINT STRING$(59,205)
390 FOR A=5 TO 22
```

```
400 LOCATE A,10:PRINT CHR$(186)
410 LOCATE A,70:PRINT CHR$(186)
420 NEXT A
430 LOCATE 23,11:PRINT STRING$(59,205)
440 LOCATE 23,10:PRINT CHR$(200):LOCATE 23,70:PRINT CHR$(188)
450 ' *****
460 ' ***** MÜŞTERİ KAYIT GİRİŞİ
470 ' *****
480 GOSUB 2740
490 COLOR 0,3:LOCATE 5,15:PRINT " ÇIKIŞ İÇİN ";:COLOR
0,7:PRINT " SIFIR ";:COLOR 7,0:COLOR 0,3:PRINT " GIRINIZ
":COLOR 3,0
500 LOCATE 8,15:INPUT "KOD NOSUNU GIRINIZ";KOD%
510 IF KOD%=0 THEN CLOSE #1:GOTO 60
520 LOCATE 9,15:LINE INPUT "ADI SOYADI.....";ADI$
530 LOCATE 10,15:LINE INPUT "ADRES 1.....";ADRES1$
540 LOCATE 11,15:LINE INPUT "ADRES 2.....";ADRES2$
550 LOCATE 12,15:LINE INPUT "TELEFON.....";TEL$
560 LOCATE 13,15:LINE INPUT "CINSI.....";CINS$
570 LOCATE 14,15:LINE INPUT "TARIHI.....";TAR$
580 LOCATE 15,15:INPUT "TUTARI.....";TUT#
590 LOCATE 16,15:INPUT "ÖDENEN.....";ODN#
600 LOCATE 17,15:LINE INPUT "AÇIKLAMA 1.....";ACIK1$
610 LOCATE 18,15:LINE INPUT "AÇIKLAMA 2.....";ACIK2$
620 KL#=TUT#-ODN#
630 LSET AD$=ADI$:LSET ADRI$=ADRES1$:LSET ADR2$=ADRES2$:LSET
C$=CINS$:LSET T$=TAR$:LSET TU$=MKD$(TUT#)
635 LSET OD$=MKD$(ODN#):LSET K$=MKD$(KL#):LSET AC1$=ACIK1$:
LSET AC2$=ACIK2$:LSET TE$=TEL$
640 COLOR 0,5:LOCATE 20,20:INPUT "GIRDIĞINIZ BİLGİLER DOĞRU
MU (E/H) ";S$:COLOR 7,0
650 IF S$="H" OR S$="h" THEN 500
660 PUT #1,KOD%:COLOR 0,7:LOCATE 22,20:PRINT "DEVAM İÇİN BİR
TUŞA BASINIZ":COLOR 7,0
670 IF INKEY$="" THEN 670
680 FOR C=7 TO 22
690 LOCATE C,15:PRINT STRING$(54," ")
700 NEXT C
710 GOTO 490
720 ' *****LISTE İSTEME BAŞLIĞI*****
730 CLS
740 COLOR 0,1:CLS
750 COLOR 0,7:LOCATE 8,10:PRINT CHR$(201):LOCATE 8,70:PRINT
CHR$(187)
760 LOCATE 8,11:PRINT STRING$(59,205)
```

```
770 FOR N=9 TO 20
780 LOCATE N,10:PRINT CHR$(186):LOCATE N,70:PRINT CHR$(186)
790 NEXT N
800 LOCATE 20,10:PRINT CHR$(200):LOCATE 20,70:PRINT CHR$(188)
810 LOCATE 20,11:PRINT STRING$(59,205):COLOR 7,0
820 COLOR 15,3:LOCATE 10,30:PRINT "MÜŞTERİ LİSTESİ":COLOR 0,1
830 COLOR 15,3:LOCATE 12,25:PRINT "1. NUMARAYA GÖRE": COLOR
0,1
840 COLOR 15,3:LOCATE 13,25:PRINT "2. İSME GÖRE":COLOR 0,1
850 COLOR 15,3:LOCATE 14,25:PRINT "3. ANA MENÜYE DÖNÜŞ":COLOR
0,1
860 COLOR 15,3:LOCATE 20,20:PRINT "LÜTFEN SEÇİMİNİZİ RAKAM
OLARAK GIRINIZ":COLOR 0,1
870 COLOR 15,3:LOCATE 16,30:INPUT "SEÇİMİNİZ";S$:COLOR 0,1
880 IF S$="1" THEN 920
890 IF S$="2" THEN 1320
900 IF S$="3" THEN COLOR 7,0:GOTO 10
910 IF S$<>" " THEN BEEP:GOTO 870
920 ' ***** KOD NOYA GÖRE BAŞLIK *****
930 CLS
940 COLOR 0,1:CLS
950 COLOR 7,1:LOCATE 4,10:PRINT CHR$(201):LOCATE 4,70:PRINT
CHR$(187)
960 LOCATE 4,11:PRINT STRING$(59,205)
970 FOR A=5 TO 22
980 LOCATE A,10:PRINT CHR$(186):LOCATE A,70:PRINT CHR$(186)
990 NEXT A
1000 LOCATE 22,10:PRINT CHR$(200):LOCATE 22,70:PRINT
CHR$(188)
1010 LOCATE 22,11:PRINT STRING$(59,205)
1020 ' *****
1030 ' ***** KAYIT OKUMA
1040 ' *****
1050 GOSUB 2740
1060 LOCATE 4,15:COLOR 0,7:PRINT " ÇIKIŞ İÇİN";:COLOR 7,0:
COLOR 0,15:PRINT " SIFIR ";:COLOR 15,0:COLOR 0,7:PRINT "
GIRINIZ ": COLOR 7,1
1070 LOCATE 6,15:INPUT "KOD NOSUNU GIRINIZ";KOD%
1080 IF KOD%=0 THEN CLOSE #1:GOTO 720
1090 LOCATE 6,15:PRINT STRING$(40," ")
1100 GET #1,KOD%
1110 ADI$=AD$:ADRES1$=ADR1$:ADRES2$=ADR2$:CINS$=C$:
TAR$=T$:TUT#=CVD(TU$):ODN#=CVD(OD$):KL#=CVD(K$):ACIK1$=AC1$:
ACIK2$=AC2$:TEL$=TE$
1120 LOCATE 6,15:PRINT "KOD NO.....";KOD%
```

```
1130 LOCATE 7,15:PRINT "ADI SOYADI.....";ADIŞ
1140 LOCATE 8,15:PRINT "ADRES 1.....";ADRES1Ş
1150 LOCATE 9,15:PRINT "ADRES 2.....";ADRES2Ş
1160 LOCATE 10,15:PRINT "TELEFON.....";TELŞ
1170 LOCATE 11,15:PRINT "CINSI.....";CINSŞ
1180 LOCATE 12,15:PRINT "TARIHI.....";TARŞ
1190 LOCATE 13,15:PRINT "TUTARI.....";TUT#
1200 LOCATE 14,15:PRINT "ÖDENEN.....";ODN#
1210 LOCATE 15,15:PRINT "KALAN.....";KL#
1220 LOCATE 16,15:PRINT "AÇIKLAMA 1.....";ACIK1Ş
1230 LOCATE 17,15:PRINT "AÇIKLAMA 2.....";ACIK2Ş
1240 COLOR 0,7:LOCATE 20,15:PRINT "DEVAM İÇİN HERHANGİ BİR
TUŞA BASINIZ":COLOR 7,1
1250 COLOR 1,0
1260 CLOSE #1
1270 IF INKEYŞ="" THEN 1270
1280 FOR A=6 TO 20 STEP 1
1290 COLOR 0,1:LOCATE A,15:PRINT STRINGŞ(50," ")
1300 NEXT A
1310 GOTO 1050
1320 ' ***** ISME GÖRE ARAMA BAŞLIĞI *****
1330 COLOR 0,1:CLS
1340 COLOR 7,1:LOCATE 4,10:PRINT CHRŞ(201):LOCATE 4,70:PRINT
CHRŞ(187)
1350 LOCATE 4,11:PRINT STRINGŞ(59,205)
1360 FOR A=5 TO 22
1370 LOCATE A,10:PRINT CHRŞ(186):LOCATE A,70:PRINT CHRŞ(186)
1380 NEXT A
1390 LOCATE 22,10:PRINT CHRŞ(200):LOCATE 22,70:PRINT
CHRŞ(188)
1400 LOCATE 22,11:PRINT STRINGŞ(59,205)
1410 ' *****
1420 ' ***** ISME GÖRE ARAMA
1430 ' *****
1440 GOSUB 2740
1450 COLOR 1,7:LOCATE 4,15:PRINT " ÇIKIŞ İÇİN 'SIFIR'
YAZINIZ ":COLOR 7,1
1460 LOCATE 6,15:INPUT "ADINI SOYADINI GIRINIZ";MŞ
1470 LOCATE 6,14:PRINT STRINGŞ(50," ")
1480 IF MŞ="0" THEN CLOSE #1:GOTO 720
1490 FOR KOD%=1 TO 999
1500 LOCATE 12,15:PRINT "LÜTFEN BEKLEYİNİZ!"
1510 GET #1,KOD%
1520 D=LEN(MŞ):MŞ=MŞ+SPACEŞ(20-D)
```



```
1530 ADIŞ=ADŞ:ADRES1Ş=ADR1Ş:ADRES2Ş=ADR2Ş:CINSŞ=CŞ:
TARŞ=TŞ:TUT#=CVD (TUŞ):ODN#=CVD (ODŞ)
1540 KL#=CVD (KŞ):ACIK1Ş=AC1Ş:ACIK2Ş=AC2Ş:TELŞ=TEŞ
1550 IF ADIŞ=MŞ THEN 1570
1560 NEXT KOD%
1570 LOCATE 12,15:PRINT STRING$(45," ")
1580 IF ADIŞ<>MŞ THEN LOCATE 12,15:PRINT "BÖYLE BİR KAYIT
YOK!":LOCATE 13,15:INPUT "LÜTFEN YENIDEN GIRINIZ..>";MŞ:
LOCATE 12,15:PRINT STRING$(45," "):LOCATE 13,15:PRINT
STRING$(45," "):GOTO 1480
1590 LOCATE 6,15:PRINT "KOD NO.....";KOD%
1600 LOCATE 7,15:PRINT "ADI SOYADI.....";ADIŞ
1610 LOCATE 8,15:PRINT "ADRES 1.....";ADRES1Ş
1620 LOCATE 9,15:PRINT "ADRES 2.....";ADRES2Ş
1630 LOCATE 10,15:PRINT "TELEFON.....";TELŞ
1640 LOCATE 11,15:PRINT "CINSI.....";CINSŞ
1650 LOCATE 12,15:PRINT "TARIHI.....";TARŞ
1660 LOCATE 13,15:PRINT "TUTARI.....";TUT#
1670 LOCATE 14,15:PRINT "ÖDENEN.....";ODN#
1680 LOCATE 15,15:PRINT "KALAN.....";KL#
1690 LOCATE 16,15:PRINT "AÇIKLAMA 1.....";ACIK1Ş
1700 LOCATE 17,15:PRINT "AÇIKLAMA 2.....";ACIK2Ş
1710 CLOSE #1
1720 COLOR 1,7:LOCATE 20,15:PRINT "DEVAM İÇİN HERHANGİ BİR
TUŞA BASINIZ ":COLOR 7,1
1730 IF INKEY$="" THEN 1730
1740 FOR A=6 TO 20 STEP 1
1750 LOCATE A,15:PRINT STRING$(50," ")
1760 NEXT A
1770 GOTO 1440
1780 ' ***** DOSYADA DEĞİŞİM İÇİN BAŞLIK*****
1790 CLS
1800 COLOR 0,1:CLS
1810 COLOR 7,1:LOCATE 4,10:PRINT CHR$(201):LOCATE 4,70:PRINT
CHR$(187)
1820 LOCATE 4,11:PRINT STRING$(59,205)
1830 FOR A=5 TO 22
1840 LOCATE A,10:PRINT CHR$(186):LOCATE A,70:PRINT CHR$(186)
1850 NEXT A
1860 LOCATE 22,10:PRINT CHR$(211):LOCATE 22,70:PRINT
CHR$(188)
1870 LOCATE 22,11:PRINT STRING$(59,205)
1880 ' *****
1890 ' ***** MÜŞTERİ KAYDINDA DEĞİŞİKLİK *****
1900 ' *****
```

```
1910 GOSUB 2740
1920 COLOR 0,7:LOCATE 6,15:INPUT "DEĞİŞİKLİK YAPILACAK
MÜŞTERİNİN KOD NUMARASI...>";KOD%:COLOR 7,1
1930 IF KOD%=0 THEN CLOSE #1:COLOR 7,0:GOTO 10
1940 LOCATE 6,15:PRINT STRING$(53," ")
1950 GET #1,KOD%
1960 ADIŞ=AD$:ADRES1Ş=ADR1$:ADRES2Ş=ADR2$:CINSŞ=C$:TARŞ=T$:
TUTŞ=CVD(TU$):ODNŞ=CVD(OD$)
1965 KLŞ=CVD(K$):ACIK1Ş=AC1$:ACIK2Ş=AC2$:TELŞ=TEŞ
1970 LOCATE 6,15:PRINT "(1)-KOD NO.....";KOD%
1980 LOCATE 7,15:PRINT "(2)-ADI SOYADI.....";ADIŞ
1990 LOCATE 8,15:PRINT "(3)-ADRES 1.....";ADRES1Ş
2000 LOCATE 9,15:PRINT "(4)-ADRES 2.....";ADRES2Ş
2010 LOCATE 10,15:PRINT "(5)-TELEFON.....";TELŞ
2020 LOCATE 11,15:PRINT "(6)-CINSI.....";CINSŞ
2030 LOCATE 12,15:PRINT "(7)-TARIHI.....";TARŞ
2040 LOCATE 13,15:PRINT "TUTARI.....";TUTŞ
2050 LOCATE 14,15:PRINT "ÖDENEN.....";ODNŞ
2060 LOCATE 15,15:PRINT "KALAN.....";KLŞ
2070 LOCATE 16,15:PRINT "(8)-AÇIKLAMA 1.....";ACIK1Ş
2080 LOCATE 17,15:PRINT "(9)-AÇIKLAMA 2.....";ACIK2Ş
2090 LOCATE 21,15:PRINT "ÇIKIŞ İÇİN [ SIFIR ] GIRİNİZ "
2100 COLOR 0,7:LOCATE 19,15:INPUT "DEĞİŞİKLİK YAPACAĞINIZ
YERİN NUMARASINI YAZIN...>";SECİM
2110 ON SECİM GOTO
2160,2170,2180,2190,2200,2210,2220,2230,2240
2120 IF SECİM=0 THEN 2260
2130 FOR A=6 TO 19 STEP 1
2140 LOCATE A,15:PRINT STRING$(50," ")
2150 NEXT A
2160 LOCATE 6,15:INPUT "KOD NO.....";KOD%:
GOTO 2250
2170 LOCATE 7,15:LINE INPUT "ADI SOYADI.....";ADIŞ:
GOTO 2250
2180 LOCATE 8,15:LINE INPUT "ADRES 1.....";
ADRES1Ş:GOTO 2250
2190 LOCATE 9,15:LINE INPUT "ADRES 2.....";
ADRES2Ş:GOTO 2250
2200 LOCATE 10,15:LINE INPUT "TELEFON.....";TELŞ:
GOTO 2250
2210 LOCATE 11,15:LINE INPUT "CINSI.....";CINSŞ:
GOTO 2250
2220 LOCATE 12,15:LINE INPUT "TARIHI.....";TARŞ:
GOTO 2250
2230 LOCATE 16,15:LINE INPUT "AÇIKLAMA 1.....";ACIK1Ş:
```

```
GOTO 2250
2240 LOCATE 17,15:LINE INPUT "AÇIKLAMA 2.....";ACIK2$:
GOTO 2250
2250 GOTO 2100
2260 '
2270 LSET AD$=ADI$:LSET ADR1$=ADRES1$:LSET ADR2$=ADRES2$:LSET
C$=CINS$:LSET T$=TAR$:LSET TU$=MKD$(TUT#):LSET
OD$=MKD$(ODN#):LSET K$=MKD$(KL#):LSET AC1$=ACIK1$:LSET
AC2$=ACIK2$:LSET TE$=TEL$
2280 PUT#1,KOD%:GOTO 1920
2290 ' ***** HESAP İŞLERİ *****
2300 CLS
2310 COLOR 0,1
2320 LOCATE 4,10:PRINT CHR$(201):LOCATE 4,70:PRINT CHR$(187)
2330 LOCATE 4,11:PRINT STRING$(59,205)
2340 FOR A=5 TO 22
2350 LOCATE A,10:PRINT CHR$(186)
2360 LOCATE A,70:PRINT CHR$(186)
2370 NEXT A
2380 LOCATE 23,11:PRINT STRING$(59,205)
2390 LOCATE 23,10:PRINT CHR$(200):LOCATE 23,70:PRINT
CHR$(188):COLOR 1,0
2400 ' ***** DÜZELTME *****
2410 GOSUB 2740
2420 LOCATE 4,15:COLOR 0,2:PRINT " ÇIKIŞ İÇİN";:COLOR 2,0:
COLOR 0,7:PRINT " SIFIR ";:COLOR 7,0:COLOR 0,2:PRINT "
GIRINIZ ": COLOR 7,0
2430 LOCATE 6,15:INPUT "KOD NOSUNU GIRINIZ";KOD%
2440 IF KOD%=0 THEN CLOSE #1:GOTO 10
2450 LOCATE 6,15:PRINT STRING$(40," ")
2460 GET #1,KOD%
2470 ADI$=AD$:ADRES1$=ADR1$:ADRES2$=ADR2$:CINS$=C$:
TAR$=T$:TUT#=CVD(TU$):ODN#=CVD(OD$):KL#=CVD(K$):ACIK1$=AC1$:
ACIK2$=AC2$:TEL$=TE$
2480 LOCATE 6,15:PRINT "KOD NO.....";KOD%
2490 LOCATE 7,15:PRINT "ADI SOYADI.....";ADI$
2500 LOCATE 8,15:PRINT "ADRES 1.....";ADRES1$
2510 LOCATE 9,15:PRINT "ADRES 2.....";ADRES2$
2520 LOCATE 10,15:PRINT "TELEFON.....";TEL$
2530 LOCATE 11,15:PRINT "CINSİ.....";CINS$
2540 LOCATE 12,15:PRINT "TARIHI.....";TAR$
2550 LOCATE 13,15:PRINT "TUTARI.....";TUT#
2560 LOCATE 14,15:PRINT "ÖDENEN.....";ODN#
2570 LOCATE 15,15:PRINT "KALAN.....";KL#
2580 LOCATE 16,15:PRINT "AÇIKLAMA 1.....";ACIK1$
```

```
2590 LOCATE 17,15:PRINT "AÇIKLAMA 2.....";ACIK2$
2600 LOCATE 19,20:PRINT "MÜŞTERİNİN SON DURUMU"
2610 COLOR 0,7:LOCATE 20,15:INPUT "ÖDENEN MIKTARI YAZINIZ";
ODEN#:COLOR 7,1
2620 KALAN#=KL#-ODEN#:ODENEN#=ODN#+ODEN#:KL#=KALAN#:
ODN#=ODENEN#
2630 LSET AD$=ADI$:LSET ADR1$=ADRES1$:LSET ADR2$=ADRES2$:LSET
C$=CINS$:LSET T$=TAR$:LSET TU$=MKD$(TUT#):LSET
OD$=MKD$(ODN#):LSET K$=MKD$(KL#):LSET AC1$=ACIK1$:LSET
AC2$=ACIK2$:LSET TE$=TEL$
2640 COLOR 0,5:LOCATE 21,20:INPUT "GİRDİĞİNİZ BİLGİLER DOĞRU
MU (E/H) ";S$:COLOR 7,0
2650 LOCATE 21,20:PRINT STRING$(50," ")
2660 IF S$="H" OR S$="h" THEN 2690
2670 PUT #1,KOD%:COLOR 0,7:LOCATE 21,20:PRINT "DEVAM İÇİN BİR
TUŞA BASINIZ":COLOR 7,0
2680 IF INKEY$="" THEN 2680
2690 FOR C=6 TO 22
2700 LOCATE C,15:PRINT STRING$(54," ")
2710 NEXT C
2720 COLOR 7,0
2730 GOTO 2420
2740 ' ***** DOSYA AÇMA *****
2750 '
2760 OPEN "R",#1,"MUSTERI.DAT",137
2770 FIELD #1,20 AS AD$,15 AS ADR1$,15 AS ADR2$,10 AS C$,10
AS T$,10 AS TU$,10 AS OD$,10 AS K$,15 AS AC1$,15 AS AC2$,7 AS
TE$
2780 RETURN
2790 CLS
2800 LOCATE 4,10:PRINT STRING$(60,219)
2810 FOR I=4 TO 23
2820 SOUND 1111,.1
2830 LOCATE I,10:PRINT CHR$(219)
2840 LOCATE I,70:PRINT CHR$(219)
2850 NEXT I
2860 LOCATE 23,10:PRINT STRING$(60,219)
2870 LOCATE 7,20:PRINT " TEŞEKKÜR EDERİM "
2880 LOCATE 15,20:PRINT "HAZIRLAYAN : " :COLOR 31,2:PRINT
" GÜVEN TANIŞ ":COLOR 7,0
2890 PRINT :PRINT :PRINT :PRINT :PRINT :PRINT :PRINT :PRINT
2900 END
```

K. BASIC PROGRAMLARININ DERLENMESİ

BASIC'te yazdığımız programları derleyerek işletim sisteminde direk çalışabilir duruma getirebiliriz. Derleme işleminden sonra programın hızı artacaktır. Derleme sonucu yazdığımız programlar **.EXE** uzantısını alır.

Basic programlarını derleyebilmek için, bir **Basic Derleyicisi** bulmamız gerekir. Basic derleyecisinde bulunması gerekli **5 Dosya** vardır. Bunlar şunlardır:

BASCOM.COM
BASCOM.LIB
BASRUN.EXE
BASRUN.LIB
LINK.EXE

Derleme olayı ise şöyle yapılır: Basic'le yazdığımız **BORDRO.BAS** adlı programı derleyelim:

Programımızı önce **ASCII** kodda saklamamız gerekmektedir. Basic ortamında **SAVE" BORDRO"**,A yazarak kayıt edelim ve **DOS**'a geçelim.

```
BASCOM BORDRO; ↵
```

Hata mesajı alınırsa, hatalar düzeltilir. Hata mesajı alınmazsa;

```
LINK BORDRO; ↵
```

Yapılarak program derlenmiş olur. Bu derleme sonucunda aynı adlı **3 Dosya** oluşur. Bunlar: **BORDRO.BAS**, **BORDRO.OBJ**, **BORDRO.EXE**'dir. Diğerlerini silerek yalnız **BORDRO.EXE** programını diskette tutmamız yeterlidir. Aynı disket içerisine **BASRUN.EXE**'yi de kopyalamamız gerekmektedir. Değilse program çalışmaz.

Not: Dosya adlarından sonra **;"** (Noktalı Virgül) koyarsak diğer sorulara cevap vermemiş oluruz. yani derleyici bizden başka bir şey istemez. Eğer **;"** koymazsak diğer sorularda dosya adını değiştirebiliriz ya da **Enter**'le boş geçerek aynı adla kabul etmiş oluruz.

Basic programlarımızı **ASCII** olarak kayıt ettikten sonra, **QUICK BASIC** altına atarak da derleyebiliriz.

SORULAR

1. Klavyeden bir malın satış fiyatı ve maliyet fiyatı girildiğinde, bu malın kar tutarını hesaplayan programı yapınız.
2. Beş işçinin ücretini hesaplayan programı yapınız.
3. 1'den 25'e kadar sayıların karelerini, küplerini ve ortalamalarını bulan programı hem sayaç ile hemde FOR...NEXT ile yapınız.
4. 10 öğrencinin 3 yazılı ve bir sözlü notunu READ/DATA ile okutarak, öğrencilerin ADINI SOYADINI, NUMARASINI, NOTLARINI ve ORTALAMALARINI yazan programı yapınız.
5. Sıralı Erişimli Dosyalarda STOK.DAT adlı dosyayı 3 modda da yalnız OPEN satırını kullanarak açınız.
6. Bir müşteri takibi için MUSTERI.DAT adlı Sıralı Erişimli bir dosya açarak müşterinin ADI SOYADI, TELEFON NO, ADRESİ, BORCU hanelerini takip edebilen bir programı Yaratmak, Kaydetmek, Okumak ve Değiştirmek amacı ile yapınız.
7. Doğrudan Erişimli Dosyalarda "Kayıt Uzunluğu 157", "Dosya Sayısı 5" olan bir programı çalıştırmak için BASIC'e nasıl geçilir?
8. MKI\$, MKS\$, MKD\$ ile CVI, CVS, CVD fonksiyonlarını karşılaştırınız. Birer örnek vererek değerleri birbirine çeviriniz.
9. 2. soruyu Doğrudan Erişimli Dosyaya göre yapınız.
10. 2. soruda yaptığımız programları MERGE ile birleştirerek tek bir program haline getiriniz. Bir Ana Menü oluşturarak ON ... GOTO ile bunları bağlayınız.
11. Bir programın başlangıcında INPUT\$ komutunu kullanarak bir şifre veriniz. Bu şifreyi en çok üç kez bilmesini isteyiniz. Şifre bilinirse program çalışsın, bilinmezse DOS'a dönsün.

ÖRNEK PROGRAMLAR

```
10 ' SAYILARI YAZIYA ÇEVİREN PROGRAM
20 INPUT "SAYI";S#
30 S#=INT(S#)
40 SAY$=STR$(S#)
50 W=LEN(SAY$)-1
60 SAY$=RIGHT$(SAY$,W)
70 PRINT "YALNIZ *** ";
80 IF (LEN(SAY$)-INT(LEN(SAY$)/3)*3)<>0 THEN SAY$="0"+SAY$:
GOTO 80
90 M=LEN(SAY$)/3
100 FOR X=M TO 1 STEP -1
110 C$=MID$(SAY$,1+(M-X)*3,3)
120 IF C$="000" THEN GOTO 440
130 IF C$="001" THEN GOTO 400
140 FOR I=1 TO 3
150 S$=MID$(C$,I,1)
160 IF I=2 THEN GOTO 280
170 IF S$="2" THEN PRINT "İKİ";
180 IF S$="3" THEN PRINT "ÜÇ";
190 IF S$="4" THEN PRINT "DÖRT";
200 IF S$="5" THEN PRINT "BEŞ";
210 IF S$="6" THEN PRINT "ALTI";
220 IF S$="7" THEN PRINT "YEDİ";
230 IF S$="8" THEN PRINT "SEKİZ";
240 IF S$="9" THEN PRINT "DOKUZ";
250 IF I=3 AND S$="1" THEN PRINT "BİR";
260 IF I=1 AND S$<>"0" THEN PRINT "YÜZ";
270 GOTO 370
280 IF S$="1" THEN PRINT "ON";
290 IF S$="2" THEN PRINT "YIRMI";
300 IF S$="3" THEN PRINT "OTUZ";
310 IF S$="4" THEN PRINT "KIRK";
320 IF S$="5" THEN PRINT "ELLİ";
330 IF S$="6" THEN PRINT "ALTMİŞ";
340 IF S$="7" THEN PRINT "YETMİŞ";
350 IF S$="8" THEN PRINT "SEKSEN";
360 IF S$="9" THEN PRINT "DOKSAN";
370 NEXT I
380 IF X=1 OR X=2 OR X=3 THEN 420
390 IF X=4 OR X=5 OR X=6 THEN 410
400 IF X<>2 THEN PRINT "BİR";
410 IF X=4 THEN PRINT "MİLYAR";
```

```
420 IF X=3 THEN PRINT "MILYON";
430 IF X=2 THEN PRINT "BIN";
440 NEXT X
450 PRINT " ***TL'DIR"
460 END
```

```
10 REM ZAR OYUNU
15 CLS
20 RANDOMIZE TIMER
30 PRINT "DEVAM İÇİN [ENTER] ÇIKIŞ İÇİN [ESC]
BASINIZ"
40 Z1=INT(RND*6)+1
50 Z2=INT(RND*6)+1
60 PRINT "ZAR=";Z1;".";Z2
70 PRINT "DEVAM MI? "
80 AŞ=INKEY$:IF AŞ="" THEN 80
90 IF AŞ=CHR$(13) THEN 40
100 IF AŞ=CHR$(27) THEN END
```

```
10 CLS
20 REM RASTGELE SAYI ÜRETME
30 RANDOMIZE TIMER
40 S=INT(RND*100)+1
50 INPUT "TUTTUĞUM SAYIYI BULUNUZ";SA
60 N=1
70 GOTO 100
80 INPUT "YENİDEN DENEYİNİZ";SA
90 N=N+1
100 PRINT :PRINT
110 IF SA>S THEN PRINT "BÜYÜK TUTTUNUZ":GOTO 80
120 IF SA<S THEN PRINT "KÜÇÜK TUTTUNUZ":GOTO 80
130 PRINT "AFERİN";N;"KEREDE BILDİNİZ"
140 INPUT "YENİDEN OYNAMAK İSTER MİSİNİZ? (E/H)...>";YŞ
150 IF YŞ="E" OR YŞ="e" THEN 10 ELSE END
```

```
10 REM :- Bilgisayarı BOOT eden GWBASIC programı (c) Mustafa DOĞAN
20 REM Programın amacı bilgisayarın nasıl kapatılıp açılacağını göstermek
30 DEF SEG=&HFFFF ' ROM tarafından kullanılan segment
40 ATLA%=&HFFF0 ' ROM'da BOOT etmek için gerekli adresin
50 REM ' saklandığı adres
60 CALL ATLA% ' QBASIC için CALL ABSOLUTE(ATLA%) yazılacak
```



```
10 ' BU PROGRAM DAKTILOGRAFI DERSI
20 ' HIZ TESTİNDE NET VURUŞU HESAPLAR
30 ' VE ŞEKİL TESTİ İLE NOT ORTALAMASINI VERİR
40 ' AÇILIŞ MENÜSÜNE DÖNMEK İÇİN BRÜT SÖZCÜĞE 999 GIRINIZ
50 CLS:KEY OFF
60 LOCATE 3,25:COLOR 0,7:PRINT " DAKTILO NOTLARININ HESABI
":COLOR 7,0
70 LOCATE 5,20:PRINT "1- LİSE BİRİNCİ SINIF I. YARIYIL"
80 LOCATE 6,20:PRINT "2- LİSE BİRİNCİ SINIF II. YARIYIL"
90 LOCATE 7,20:PRINT "3- LİSE İKİNCİ SINIF I. YARIYIL"
100 LOCATE 8,20:PRINT "4- LİSE İKİNCİ SINIF II. YARIYIL"
110 LOCATE 9,20:PRINT "5- PROGRAMDAN ÇIKIŞ"
120 LOCATE 14,25:COLOR 0,7:INPUT "SEÇENEĞİNİZ....>
",S$:COLOR 7,0
130 S=VAL(S$)
140 IF S<1 OR S>5 THEN BEEP:BEEP:GOTO 120
150 ON S GOTO 160,400,630,890,1150
160 CLS
170 PRINT "LİSE BİRİNCİ SINIFLAR DAKTILOGRAFI DERSİ I.
YARIYIL NOT HESABI"
180 PRINT :PRINT
190 INPUT "BİR YANLIŞ KAÇ DOĞRUYU GÖTÜRECEK";G
200 CLS
210 INPUT "BRÜT SÖZCÜĞÜ GIRINIZ";B
220 IF B=999 THEN 50
230 INPUT "YANLIŞ SAYISINI GIRINIZ";Y
240 NET=B-(Y*G)
250 PRINT "NET VURUŞU :";NET;" DIR"
260 PRINT "ALDIĞI NOT :";
270 IF NET<=10 THEN PRINT "1"
280 IF NET>=11 AND NET<=25 THEN PRINT "2"
290 IF NET>=26 AND NET<=40 THEN PRINT "3"
300 IF NET>=41 AND NET<=55 THEN PRINT "4"
310 IF NET>=56 AND NET<=75 THEN PRINT "5"
320 IF NET>=76 AND NET<=85 THEN PRINT "6"
330 IF NET>=86 AND NET<=100 THEN PRINT "7"
340 IF NET>=101 AND NET<=115 THEN PRINT "8"
350 IF NET>=116 AND NET<=120 THEN PRINT "9"
360 IF NET>=121 THEN PRINT "10"
370 PRINT :PRINT :PRINT :PRINT "İŞ BITİMİNDE HERHANGİ BİR
TUŞA BASINIZ"
380 IF INKEY$="" THEN 380
390 GOTO 200
400 CLS
```

```
410 PRINT "LİSE BİRİNCİ SINIFLAR DAKTİLOGRAFI DERSİ II.
YARIYIL NOT HESABI"
420 INPUT "BİR YANLIŞ KAÇ DOĞRUYU GÖTÜRECEK";G
430 CLS
440 INPUT "BRÜT SÖZCÜĞÜ GIRINIZ";B
450 IF B=999 THEN 50
460 INPUT "YANLIŞ SAYISINI GIRINIZ";Y
470 NET=B-(Y*G)
480 PRINT "NET VURUŞU :";NET;" DIR"
490 PRINT "ALDIĞI NOT :";
500 IF NET<=20 THEN PRINT "1"
510 IF NET>=21 AND NET<=35 THEN PRINT "2"
520 IF NET>=36 AND NET<=50 THEN PRINT "3"
530 IF NET>=51 AND NET<=65 THEN PRINT "4"
540 IF NET>=66 AND NET<=80 THEN PRINT "5"
550 IF NET>=81 AND NET<=95 THEN PRINT "6"
560 IF NET>=96 AND NET<=110 THEN PRINT "7"
570 IF NET>=111 AND NET<=125 THEN PRINT "8"
580 IF NET>=126 AND NET<=140 THEN PRINT "9"
590 IF NET>=141 THEN PRINT "10"
600 PRINT :PRINT :PRINT :PRINT "İŞ BITİMİNDE HERHANGİ BİR
TUŞA BASINIZ"
610 IF INKEY$="" THEN 610
620 GOTO 430
630 CLS
640 PRINT "LİSE İKİNCİ SINIFLAR DAKTİLOGRAFI DERSİ I. YARIYIL
NOT HASABI"
650 INPUT "BİR YANLIŞ KAÇ DOĞRUYU GÖTÜRECEK";G
660 CLS
670 INPUT "BRÜT SÖZCÜĞÜ GIRINIZ";B
680 IF B=999 THEN 50
690 INPUT "YANLIŞ SAYISINI GIRINIZ";Y
700 INPUT "ŞEKİL NOTU";SN
710 NET=B-(Y*G)
720 PRINT "NET VURUŞU :";NET;" DIR"
730 IF NET<=25 THEN HN=10
740 IF NET>=26 AND NET<=40 THEN HN=20
750 IF NET>=41 AND NET<=55 THEN HN=30
760 IF NET>=56 AND NET<=70 THEN HN=40
770 IF NET>=71 AND NET<=85 THEN HN=50
780 IF NET>=86 AND NET<=100 THEN HN=60
790 IF NET>=101 AND NET<=115 THEN HN=70
800 IF NET>=116 AND NET<=130 THEN HN=80
810 IF NET>=131 AND NET<=150 THEN HN=90
820 IF NET>=151 THEN HN=100
```

```
830 ONOT=(HN+SN)/2
840 PRINT "HIZ NOTU :";HN
850 PRINT "ORTALAMA NOTU ";ONOT
860 PRINT :PRINT :PRINT :PRINT "İŞ BITİMİNDE HERHANGİ BİR
TUŞA BASINIZ"
870 IF INKEY$="" THEN 870
880 GOTO 660
890 CLS
900 PRINT "LİSE İKİNCİ SINIFLAR DAKTİLOGRAFI DERSİ II.
YARIYIL NOT HESABI"
910 INPUT "BİR YANLIŞ KAÇ DOĞRUYU GÖTÜRECEK";G
920 CLS
930 INPUT "BRÜT SÖZCÜĞÜ GIRINIZ";B
940 IF B=999 THEN 50
950 INPUT "YANLIŞ SAYISINI GIRINIZ";Y
960 INPUT "ŞEKİL NOTU";SN
970 NET=B-(Y*G)
980 PRINT "NET VURUŞU :";NET;" DIR"
990 IF NET<=35 THEN HN=10
1000 IF NET>=36 AND NET<=50 THEN HN=20
1010 IF NET>=51 AND NET<=65 THEN HN=30
1020 IF NET>=66 AND NET<=80 THEN HN=40
1030 IF NET>=81 AND NET<=95 THEN HN=50
1040 IF NET>=96 AND NET<=115 THEN HN=60
1050 IF NET>=116 AND NET<=130 THEN HN=70
1060 IF NET>=131 AND NET<=150 THEN HN=80
1070 IF NET>=151 AND NET<=170 THEN HN=90
1080 IF NET>=171 THEN HN=100
1090 ONOT=(HN+SN)/2
1100 PRINT "HIZ NOTU :";HN
1110 PRINT "ORTALAMA NOTU ";ONOT
1120 PRINT :PRINT :PRINT :PRINT "İŞ BITİMİNDE HERHANGİ BİR
TUŞA BASINIZ"
1130 IF INKEY$="" THEN 1130
1140 GOTO 920
1150 END
```

```
10 ' HAZIRLAYAN : GÜVEN TANIŞ
20 ' BU PROGRAM KOLLU OYUN MAKİNESİ'DİR.
30 ' ŞEKİLLER RASTGELE ÜRETİLMEKTEDİR
40 ' AMAÇ AYNI ŞEKİLLER YANYANA GELDİĞİNDE KAZANMAKTIR
50 ' &&& ŞEKİLLERDEN İKİSİ YANYANA GELİRSE 2 KATI ÜÇÜ YANYANA
GELİRSE 10 KATI
60 ' %%% ŞEKİLLERDEN İKİSİ YANYANA GELİRSE 4 KATI ÜÇÜ YANYANA
GELİRSE 20 KATI
70 ' ### ŞEKİLLERDEN İKİSİ YANYANA GELİRSE 6 KATI ÜÇÜ YANYANA
GELİRSE 30 KATI
80 ' YANYANA ŞEKİL YOK İSE PARA VERMEZ
90 CLS: CLEAR: RESET: KEY OFF: SCREEN 0,0,0
100 LOCATE 5,30: COLOR 0,7: PRINT " KOLLU OYUN MAKİNESİ ": COLOR
7,0
110 DEFINT A-Z: RANDOMIZE TIMER
120 'A=100:B=500:C=1000
130 'AB$=STR$(A):BB$=STR$(B):CB$=STR$(C)
140 AB$=" &&& ":BB$=" %%% ":CB$=" ### "
150 TOPP#=50000!
160 LOCATE 8,10: PRINT CHR$(219)
170 LOCATE 8,11: PRINT STRING$(61,219)
180 LOCATE 8,70: PRINT CHR$(219)
190 FOR X=9 TO 18
200 LOCATE X,10: PRINT CHR$(219);CHR$(219):LOCATE X,70: PRINT
CHR$(219);CHR$(219)
210 NEXT X
220 LOCATE 19,10: PRINT CHR$(219)
230 LOCATE 19,11: PRINT STRING$(61,219)
240 LOCATE 19,70: PRINT CHR$(219)
250 LOCATE 12,20: PRINT AB$:LOCATE 12,38: PRINT BB$:LOCATE
12,55: PRINT CB$
260 LOCATE 2,20: PRINT BB$
270 LOCATE 3,20: PRINT "10 KATI"
280 LOCATE 2,38: PRINT AB$
290 LOCATE 3,38: PRINT "20 KATI"
300 LOCATE 2,55: PRINT CB$
310 LOCATE 3,55: PRINT "30 KATI"
320 LOCATE 15,39: PRINT SPACE$(20)
330 LOCATE 18,39: PRINT SPACE$(15)
340 LOCATE 15,20: PRINT "TOPLAM PARANIZ : ";: PRINT USING
"###,###,###";TOPP#
350 IF TOPP#>75000000# THEN LOCATE 22,32: COLOR 0,7: PRINT "
KASA İFLAS ETTİ ! ": COLOR 7,0: END
360 LOCATE 18,25: INPUT "PARA GIRINIZ";A$:QW#=VAL(A$)
```

```
370 IF QW#<1 THEN BEEP:BEEP:LOCATE 22,20:COLOR 31,0:PRINT "
YANLIŞ GİRİŞ YAPTINIZ YENİDEN GIRINIZ ":COLOR 7,0:GOTO 330
380 IF QW#>1E+07 THEN LOCATE 22,20:COLOR 31,0:PRINT "
10.000.000 TL'DAN BÜYÜK GİREMEZSİNİZ ":COLOR 7,0:GOTO 320
390 IF QW#>TOPP# THEN LOCATE 22,20:COLOR 31,0:PRINT "BU KADAR
PARANIZ YOK YENİDEN GIRINIZ":COLOR 7,0:GOTO 320
400 LOCATE 22,20:PRINT SPACE$(40)
410 FOR XC=1 TO 20
420 A=(RND*2)+1
430 B=(RND*2)+1
440 C=(RND*2)+1
450 SOUND 2000,1
460 IF A=1 THEN A1=A:LOCATE 12,20:PRINT AB$
470 IF A=2 THEN A2=A:LOCATE 12,20:PRINT BB$
480 IF A=3 THEN A3=A:LOCATE 12,20:PRINT CB$
490 IF B=1 THEN A4=B:LOCATE 12,38:PRINT AB$
500 IF B=2 THEN A5=B:LOCATE 12,38:PRINT BB$
510 IF B=3 THEN A6=B:LOCATE 12,38:PRINT CB$
520 IF C=1 THEN A7=C:LOCATE 12,55:PRINT AB$
530 IF C=2 THEN A8=C:LOCATE 12,55:PRINT BB$
540 IF C=3 THEN A9=C:LOCATE 12,55:PRINT CB$
550 NEXT XC
560 PARA#=0
570 IF A=2 AND B=2 AND C=2 THEN PARA#=QW#*10:GOTO 660
580 IF A=1 AND B=1 AND C=1 THEN PARA#=QW#*20:GOTO 660
590 IF A=3 AND B=3 AND C=3 THEN PARA#=QW#*30:GOTO 660
600 IF A=2 AND B=2 THEN PARA#=QW#*2:GOTO 660
610 IF A=1 AND B=1 THEN PARA#=QW#*4:GOTO 660
620 IF A=3 AND B=3 THEN PARA#=QW#*6:GOTO 660
630 IF B=2 AND C=2 THEN PARA#=QW#*2:GOTO 660
640 IF B=1 AND C=1 THEN PARA#=QW#*4:GOTO 660
650 IF B=3 AND C=3 THEN PARA#=QW#*6:GOTO 660
660 IF PARA#>0 THEN TOPP#=TOPP#+PARA#
670 IF PARA#=0 THEN TOPP#=TOPP#-QW#
680 LOCATE 15,20:PRINT "TOPLAM PARANIZ : ";:PRINT USING
"###,###,###";TOPP#
690 IF TOPP#=0 THEN 730
700 IF PARA#=0 THEN LOCATE 22,20:COLOR 31,0:PRINT "ÜZGÜNÜM
KAYBETTİNİZ YENİDEN GIRINIZ":COLOR 7,0
710 GOTO 320
720 FOR W=1 TO 100:NEXT W
730 LOCATE 15,35:PRINT SPACE$(20)
740 FOR D=1 TO 50
750 LOCATE 20,20:COLOR 0,15:PRINT " KUMAR ÇOK KÖTÜ BİR
ALIŞKANLIKTIR ":COLOR 7,0
```

```
760 LOCATE 23,23:COLOR 31,0:PRINT " BÜTÜN PARANIZI
KAYBETTİNİZ ":COLOR 7,0
770 NEXT D:CLS
780 END
```

```
10 REM          :-) Ekranda ufak bir gezinti (c) Mustafa DOĞAN
20 REM          Ok tuşlarıyla ekranda, ekranı silmeden blok gezdirin
30 REM          Programın amacı hem ok tuşlarını kullanmayı,
40 REM          hem de ekran adresini direkt kullanmayı öğretmektir.
50 YATAY=35      ' Yatay koordinat ayarlanıyor
60 DIKEY=13      ' Dikey koordinat ayarlanıyor
70 RENK=&H70     ' Başlangıç rengi siyah zemine beyaz
80 UZUNLUK=10    ' Blokun boyu
90 DEF SEG=&HB000 ' Burası ÖNEMLİ Renkli makinalarda &HB800 olmalı
100 OFFSET=0     ' Bu satır aslında olmasa da olur. Süs olsun diye
110 GOSUB 410    ' Ekranı çiz.
120 REM *****
130 TUS$=INKEY$:IF TUS$="" THEN 130      ' Tuş okuttur
140 IF TUS$=CHR$(27) THEN DEF SEG:END    ' Tuşları değerlendir
150 IF TUS$=CHR$(0)+"H" THEN 210        ' YUKARI ok tuşu
160 IF TUS$=CHR$(0)+"K" THEN 260        ' SOL ok tuşu
170 IF TUS$=CHR$(0)+"M" THEN 310        ' SAĞ ok tuşu
180 IF TUS$=CHR$(0)+"P" THEN 360        ' AŞAĞI ok tuşu
190 GOTO 130      ' Başa dön
200 REM *****
210 RENK=7:GOSUB 410      ' Eski bloku sil
220 IF DIKEY>1 THEN DIKEY=DIKEY-1      ' Yeni koordinatları ayarla
230 RENK=&H70:GOSUB 410      ' Tekrar bloku çiz
240 GOTO 130      ' Başa dön
250 REM *****
260 RENK=7:GOSUB 410
270 IF YATAY>1 THEN YATAY=YATAY-1
280 RENK=&H70:GOSUB 410
290 GOTO 130
300 REM *****
310 RENK=7:GOSUB 410
320 IF YATAY<81-UZUNLUK THEN YATAY=YATAY+1
330 RENK=&H70:GOSUB 410
340 GOTO 130
350 REM *****
360 RENK=7:GOSUB 410
370 IF DIKEY<25 THEN DIKEY=DIKEY+1
```

```
380 RENK=&H70:GOSUB 410
390 GOTO 130
400 REM *****
410 OFSET=( (DIKEY-1)*80+(YATAY-1))*2 ' Ekranın gerçek adresi
420 FOR DONGU=1 TO UZUNLUK*2-1 STEP 2
430 POKE OFSET+DONGU,RENK ' Bloku adrese yaz
440 NEXT
450 RETURN

10 REM          :-) Joystick test programı (c) Mustafa DOĞAN
20 REM  Ekranı X ve Y koordinatlarıyla ateş tuşlarının durumunu gösterir.
30 CLS
40 STRIG(0) ON          ' Üst ateş tuşu
50 STRIG(4) ON          ' Alt ateş tuşu
60 WHILE INKEY$=""
70 LOCATE 1,1:PRINT "X=";STICK(0),"Y=";STICK(1)," "
80 PRINT "              "
90 PRINT "              "
100 ON STRIG(0) GOSUB 130 ' Üst tuş böyle kontrol edilir
110 ON STRIG(4) GOSUB 150 ' Alt tuş da böyle kontrol edilir
120 WEND
130 LOCATE 2,1:PRINT "Üst Ateş tuşuna basıldı "
140 RETURN
150 LOCATE 3,1:PRINT "Alt Ateş tuşuna basıldı "
160 RETURN

10 REM          :-) Değişik bir ses çıkartma yöntemi (c) Mustafa DOĞAN
20 KONTROL=INP(&H61) ' Hoparlörü kontrol eden byte yedekleniyor
30 GONDER=KONTROL
40 FOR DONGU=1 TO 1000 ' Döngü
50 GONDER=GONDER XOR 2 ' Hoparlör açıksa kapat, kapalıysa aç
60 OUT &H61,GONDER ' Hoparlöre kontrol byte'i gönderiliyor
70 NEXT
80 OUT &H61,KONTROL ' Program hoparlör kontrol byte'ını geri koyar
```

```
10 REM          :-) Kendi halinde bir diagnostic programı (c) Mustafa DOĞAN
20 REM          Program, bilgisayarla ilgili bilgileri ROM BIOS Data Area'dan
30 REM          almaktadır. (Belleğin ilk 1 KB.'ı).
40 REM          0:411h adresinde bilgisayarın donanım listesi bulunur.
50 REM          Biz de bu adresi kullandık.
60 DEF SEG=0          ' ROM bilgi alanına bakacağız
70 LISTE1=PEEK(&H410) ' Donanım listesinin yarısının adresi
80 LISTE2=PEEK(&H411) ' Donanım listesinin diğer yarısının adresi
90 CLS
100 PRINT "Bilgisayarınızın içindekilere bir göz atalım:":
PRINT
110 PRINT "Bilgisayara bağlı en az bir sürücü ";
120 IF (LISTE1 AND 1)=1 THEN PRINT "var." ELSE PRINT "bile
yok." ' 0. bit
130 PRINT "Matematik yardımcı işlemci ";
140 IF (LISTE1 AND 2)=2 THEN PRINT "var." ELSE PRINT "yok."
' 1. bit
150 PRINT "Ekran kartı ";          ' 2 ve 3. bitler
160 IF (LISTE1 AND 48)=32 THEN PRINT "renkli." ELSE PRINT
"siyah-beyaz."
170 IF (LISTE1 AND 192)=0 THEN PRINT "1 " ELSE PRINT "2 ";
' 6 ve 7. bitler
180 PRINT "adet disket sürücünüz var."
190 IF (LISTE2 AND 14)=2 THEN PRINT "1 ";          ' 1, 2, ve 3. bitler
200 IF (LISTE2 AND 14)=4 THEN PRINT "2 ";
210 PRINT "adet seri portunuz var."
220 PRINT "Oyun adaptörünüz ";
230 IF (LISTE2 AND 16)=16 THEN PRINT "var." ELSE PRINT "yok."
' 4. bit
240 PRINT "Yazıcı bağlamak için ";
250 IF (LISTE2 AND 192)=64 THEN PRINT "1 " ELSE PRINT "2 ";
' 6 ve 7. bit
260 PRINT "adet paralel portunuz var.":PRINT

10 CLS
20 DEFINT A-Z
30 INPUT "GÜNÜ-AYI-YILI GIR ";G,A,Y
40 IF A=1 THEN A=11 :Y=Y-1:GOTO 70
50 IF A=2 THEN A=12 :Y=Y-1:GOTO 70
60 A=A-2
70 I=VAL(MID$(STR$(Y),2,2)):S=VAL(MID$(STR$(Y),4,2))
80 X=INT(A*2.6-.2):Y=INT(S/4):Z=INT(I/4)
90 W=X+Y+Z+G+S-2*I:T=W MOD 7
100 IF T=0 THEN GUN$="PAZAR"
```



```

110 IF T=1 THEN GUNŞ="PAZARTESI"
120 IF T=2 THEN GUNŞ="SALI"
130 IF T=3 THEN GUNŞ="ÇARŞAMBA"
140 IF T=4 THEN GUNŞ="PERŞEMBE"
150 IF T=5 THEN GUNŞ="CUMA"
160 IF T=6 THEN GUNŞ="CUMARTESI"
170 PRINT GUNŞ

10 REM          :-) Zıplayan yüz programı (c) Mustafa DOĞAN
20 REM          Eğlenceli bir grafik uygulaması.
30 REM          Program siyah-beyazdır. Renklendirmeyi deneyin. (Çok basittir)
40 REM          Eğer bunu yapabildinizse birden fazla yüzü aynı anda zıplatın.
50 REM          Bu durumda zıplayan yüzlerin yavaşlaması normaldir.
60 REM Programı yazarken kendi grafik kartınıza göre koordinatları ayarlayın
70 REM      :-< Hercules için yazıldı. (720x348 piksel ekran boyutları için)
80 REM      :-< Diğer grafik kartları için koordinatlar değiştirilmeli.
90 REM      :-< GWBASIC Screen 3'ü desteklemeyebilir. DOS'un QBASIC'i ile RUN
100 SCREEN 3          ' Screen 3 Hercules görüntü kartları içindir.
110 DIM IMAGE%(1000)  ' Diğer görüntü kartları için bu modu seçmeyin
120 RANDOMIZE TIMER
130 MAXX=720          ' Grafik kartının maximum X koordinatı
140 MAXY=348          ' Grafik kartının maximum Y koordinatı
150 POSX1=INT(RND*(MAXX-50))+2
160 POSY1=INT(RND*(MAXY-50))+2
170 CIRCLE (POSX1+22,POSY1+22),20          'Yüz çiziliyor
180 CIRCLE (POSX1+22,POSY1+22),4
190 CIRCLE (POSX1+22-4,POSY1+22-6),3
200 CIRCLE (POSX1+22+4,POSY1+22-6),3
210 CIRCLE (POSX1+22,POSY1+22),10,,3.1415 'Ağız çiziliyor
220 GET (POSX1,POSY1)-(POSX1+44,POSY1+44),IMAGE% 'Ekrandaki
230 REM          görüntü bir diziye aktarılıyor
240 POSX2=INT(RND*(MAXX-50))+2:IF POSX2<POSX1 THEN YONX=-1
250 POSY2=INT(RND*(MAXY-50))+2:IF POSY2<POSY1 THEN YONY=-1
260 LINE (0,0)-(MAXX-1,MAXY-1),,B
270 REM *****
280 WHILE INKEY$=""
290 IF YONX=1 THEN GOSUB 340 ELSE GOSUB 440'X koordinatı ayarlatılıyor
300 IF YONY=1 THEN GOSUB 390 ELSE GOSUB 490'Y koordinatı ayarlatılıyor
310 PUT (POSX1,POSY1),IMAGE%,PSET
320 WEND
330 REM *****
340 REM **** X koordinatı ayarlanıyor
350 IF POSX1<POSX2 THEN POSX1=POSX1+1 ELSE YONX=-1

```

```
360 IF POSX1=POSX2 THEN POSX2=INT(RND*(MAXX-50))+2
370 RETURN
380 REM *****
390 REM **** Y koordinatı ayarlanıyor
400 IF POSY1<POSY2 THEN POSY1=POSY1+1 ELSE YONY=-1
410 IF POSY1=POSY2 THEN POSY2=INT(RND*(MAXY-50))+2
420 RETURN
430 REM *****
440 REM **** X koordinatı ayarlanıyor
450 IF POSX1>POSX2 THEN POSX1=POSX1-1 ELSE YONX=1
460 IF POSX1=POSX2 THEN POSX2=INT(RND*(MAXX-50))+2
470 RETURN
480 REM *****
490 REM **** Y koordinatı ayarlanıyor
500 IF POSY1>POSY2 THEN POSY1=POSY1-1 ELSE YONY=1
510 IF POSY1=POSY2 THEN POSY2=INT(RND*(MAXY-50))+2
520 RETURN
530 END

10 CLS
20 PRINT
30 PRINT
40 PRINT
50 PRINT "PASTA GRAFIK PROGRAMI"
60 PRINT
70 PRINT "HAZIRLAYAN: MURAT YILDIRIMOĞLU"
80 YŞ=INKEY$
90 IF YŞ="" THEN 80
100 CLS
110 PRINT
120 PRINT
130 PRINT "BU PROGRAM EN ÇOK 15 ADET VERİ GIRILDIĞINDE"
140 PRINT "BU VERİLERİ PASTA GRAFIKLERİ ŞEKLİNE"
150 PRINT "DÖNÜŞTÜRÜR.ÇALIŞMAYA BAŞLADIĞINDA VERİLERİ TEK
TEK"
160 PRINT "GİRİNİZ. VERİLERİN SONUNU GÖSTERMEK İÇİN 0
DEĞERİNİ"
170 PRINT "GİRİNİZ."
180 PRINT
190 PRINT "HERHANGİ BİR TUŞA BASINIZ."
200 YŞ=INKEY$
210 IF YŞ="" THEN 200
220 CLS
230 PRINT
240 PRINT
```

```
250 DIM VERI(15),AD$(15)
255 I=1:SUM=0
257 INPUT "TABLO İÇİN BAŞLIK GIRINIZ ",BAS$
258 PRINT
259 PRINT
260 INPUT "VERİNİN ADINI GIRINIZ.";AD$(I)
270 IF AD$(I)="0" THEN 310
275 INPUT "VERİYİ GIRINIZ.";VERI(I)
280 SUM=SUM+VERI(I)
290 I=I+1
300 GOTO 260
310 FOR K=1 TO I
320 VERI(K)=VERI(K)/SUM
330 NEXT K
340 CLS
350 SCREEN 2
355 ARG=0
357 LINE (360,150)-(510,150)
358 PI=2*3.141592653#
360 FOR K=1 TO I-1
370 ACI=VERI(K)*PI
375 START=ARG
376 SON=ARG+ACI
377 IF SON>6.28 THEN SON=6.28
380 FOR H=START TO SON STEP .005
400 PSET (360+150*COS(H),150-100*SIN(H))
420 NEXT H
440 LINE (360,150)-(360+150*COS(SON),150-100*SIN(SON))
441 YARI=(SON-START)/2+START
442 XEL=360+180*COS(YARI)
444 YEL=150-122*SIN(YARI)
454 LINE (360+150*COS(YARI),150-100*SIN(YARI))-(XEL,YEL),10
456 SAT=80*XEL/720
458 SUT=25*YEL/340
460 LOCATE SUT,SAT:PRINT AD$(K)
470 ARG=ARG+ACI
475 IF K<>1 THEN 480
478 PAINT (360+130*COS(YARI),150-(SIN(YARI)*(130/15))),10
480 NEXT K
500 LOCATE 1,1
520 PRINT " ";BAS$
540 LOCATE 3,1
560 FOR M=1 TO I-1
580 PRINT M;AD$(M);"=";100*VERI(M)
600 NEXT M
```

```
10 CLS:KEY OFF
20 CLEAR:SCREEN 0,0
30 F=1:I=1
40 LOCATE 25,25:PRINT "HAZIRLAYAN: GÜVEN TANIŞ"
50 FOR A=1 TO 10
60 FOR B=1 TO 10
70 C=A*B
80 LOCATE I,F:PRINT A;"*";B;"=";C
90 I=I+1
100 IF B=10 THEN F=F+16
110 IF I=11 THEN I=1
120 IF I=23 THEN I=13
130 IF A=5 AND B=10 THEN PRINT :COLOR 0,7:PRINT
"*****"
ÇARPIM TABLOSU "*****":COLOR 7,0
140 NEXT B
150 IF A=5 THEN I=13
160 IF A=5 THEN F=1
170 NEXT A
180 END
```

Telefon Rehberi Programı:

```
10 CLS: CLEAR ' Bütün deĞişkenler boşalıyor ve sıfırlanıyor
20 LOCATE 5,32: PRINT "TELEFON REHBERİ"
30 LOCATE 10,25: PRINT "1- AD SOYADA GÖRE ARAMA"
40 LOCATE 11,25: PRINT "2- TELEFON NUMARASINA GÖRE ARAMA"
50 LOCATE 12,25: PRINT "3- PROGRAMDAN ÇIKIŞ"
60 LOCATE 14,30: INPUT "SEÇİMİNİZ...>"; S
70 IF S<1 OR S>3 THEN BEEP: BEEP: GOTO 60
80 IF S=1 THEN 110
90 IF S=2 THEN 170
100 IF S=3 THEN END
110 CLS
120 INPUT "ARADIĞINIZ ABONENİN ADINI SOYADINI GIRINIZ"; ADŞ
130 READ TİŞŞ, TEVŞ, AKŞ, ADİŞ, ADRESŞ, SEHIRŞ
140 IF ADŞ=ADİŞ THEN 230
150 IF ADŞ="000" OR ADİŞ="000" THEN LOCATE 12,20: PRINT "BÖYLE
BİR ABONE YOK": GOTO 300
160 GOTO 130
170 CLS
180 INPUT "ARADIĞINIZ TELEFON NUMARASINI GIRINIZ"; TELŞ
190 READ TİŞŞ, TEVŞ, AKŞ, ADİŞ, ADRESŞ, SEHIRŞ
200 IF TİŞŞ=TELŞ OR TEVŞ=TELŞ THEN 230
210 IF TELŞ="000" OR TİŞŞ="000" OR TEVŞ="000" THEN LOCATE
12,20: PRINT "BOYLE BİR ABONE YOK": GOTO 300
220 GOTO 190
230 CLS
240 LOCATE 5,5: PRINT "ADI SOYADI.....: "; ADİŞ
250 LOCATE 7,5: PRINT "TELEFON NUMARASI. [IŞ]..: "; TİŞŞ
260 LOCATE 9,5: PRINT "TELEFON NUMARASI. [EV]..: "; TEVŞ
270 LOCATE 11,5: PRINT "ALAN KODU.....: "; AKŞ
280 LOCATE 13,5: PRINT "ADRESİ.....: "; ADRESŞ
290 LOCATE 15,5: PRINT "ŞEHİR.....: "; SEHIRŞ
300 LOCATE 20,20: INPUT "YENİDEN ARAMAK İSTER MİSİNİZ?
(E/H)"; YŞ
310 IF YŞ="E" OR YŞ="e" THEN 10 ELSE END
320 DATA "YOK", "7134143", "332", "GÜVEN TANIŞ", "DALMAZ MAH.
FUZULİ SO. NO:1/1", "EREĞLİ, KONYA"
330 DATA "YOK", "3164385", "286", "İSMAİL ŞEN", "SAKARYA MAH.
HIZLI GAZETECİ SO. 12", "BIGA, ÇANAKKALE"
340 DATA "YOK", "4286419", "312", "MUKADDES PERİN", "ALİ BABA
SO.14/12", "A. AYRANCI, ANKARA"
350 DATA "5134239", "5132035", "358", "DR. GÜLAY REİS",
"CUMHURİYET CAD.", "MERZİFON, AMASYA"
360 DATA "YOK", "2306146", "212", "MUAZZEZ MENEMENCİOĞLU",
```

```
"KUVAYI MİLLİYE CAD.146/8","ŞİŞLİ, İSTANBUL"
370 DATA "YOK","3162869","286","RUHI ÖZGAN","HAMDI BEY MAH.
TEPE SOK. NO 78","BIGA, ÇANAKKALE"
380 DATA "3162616","3163660","286","HAYRETTİN PİŞKİN","YENİ
BELEDİYE PASAJI KAT:2 NO:3-4","BIGA, ÇANAKKALE"
390 DATA "YOK","7511130","258","HASAN HELVACI","YUKARI SEYİT
KÖYÜ","ÇAL, DENİZLİ"
400 DATA "3161323","3161781","266","FAHRI KARAPINAR",
"BELEDİYE PASAJI 19","BIGA, ÇANAKKALE"
410 DATA "1456","3165975","286","AHMET SAAT","SAKARYA MAH.
BAHÇE SOK. NO:13","BIGA, ÇANAKKALE"
420 DATA "3161035","3164330","286","RAMAZAN ŞEKER","BIGA
TIC.LIS.MD.", "BIGA, ÇANAKKALE"
430 DATA "6612210","6612412","222","İBRAHİM DUMAN",
"SARICAKAYA MİLMÜDÜRLÜĞÜ","SARICAKAYA, ESKİŞEHİR"
1000 DATA "000","000","000","000","000","000"
```

Bu programı çalıştırdığımızda, DATA'lara İsm'e göre ya da Telefon Numarasına göre erişebiliriz. Aynı zamanda adresleri de buradan bulabiliriz.

Şimdi de öğrencilerin not durumunu bildiren programı yapalım:

```
10 CLS: CLEAR ' Bütün Sayısal Değişkenler sıfırlanıyor ve Alfayısal Değ. Boşalıyor.
20 LOCATE 10,15: PRINT "1- NUMARAYA GÖRE ARAMA"
30 LOCATE 11,15: PRINT "2- LİSTE DÖKÜMÜ"
40 LOCATE 12,15: PRINT "3- ÇIKIŞ"
50 LOCATE 14,15: COLOR 0,7: INPUT "SEÇİMİNİZ"; SC: COLOR 7,0
60 IF SC=1 THEN 90
70 IF SC=2 THEN 250
80 IF SC=3 THEN END
90 CLS
100 INPUT "ÖĞRENCİ NUMARASI"; NMŞ
110 PRINT "NUMARASI", "ADI SOYADI", "YAZILILAR", "SÖZLÜ",
"ORTALAMA"
120 PRINT "=====", "=====", "=====", "=====",
"====="
130 PRINT
140 READ NUMŞ, ADŞ, Y1, Y2, Y3, SZL
150 IF NMŞ="000" OR NUMŞ="000" THEN PRINT "BU NUMARADA BİR
ÖĞRENCİ YOK": GOTO 190
160 ORT=(Y1+Y2+Y3+SZL)/4
170 IF NMŞ=NUMŞ THEN 180 ELSE 140
180 PRINT NUMŞ, ADŞ, Y1; Y2; Y3, SZL, ORT
190 PRINT : PRINT : PRINT : INPUT "YENİDEN SORACAK MISINIZ?...
(E/H) >"; YŞ
```

```
200 IF YŞ="E" OR YŞ="e" THEN 100 ELSE 10
210 CLS
220 PRINT :PRINT :PRINT "GÖRÜNÜZ VE BİR TUŞA BASINIZ"
230 IF INKEYŞ="" THEN 230
240 GOTO 10
250 CLS
260 PRINT "NUMARASI", "ADI SOYADI", "YAZILILAR", "SÖZLÜ",
"ORTALAMA"
270 PRINT "=====", "=====", "=====", "=====",
"====="
280 PRINT
290 READ NUMŞ, ADŞ, Y1, Y2, Y3, SZL
300 IF NMŞ="000" OR NUMŞ="000" THEN 220
310 ORT=(Y1+Y2+Y3+SZL)/4
320 PRINT NUMŞ, ADŞ, Y1; Y2; Y3, SZL, ORT
330 GOTO 290
340 DATA "78", "CANAN ŞAHİN", 4, 9, 5, 7
350 DATA "55", "VILDAN TAŞKIN", 4, 6, 8, 5
360 DATA "234", "SIBEL ÇETİN", 8, 3, 7, 7
370 DATA "165", "KASIM BİLİK", 6, 4, 5, 8
380 DATA "270", "ŞERİFE KORKMAZ", 7, 6, 8, 7
390 DATA "477", "ÇİLEM KIZILOK", 7, 9, 8, 8
400 DATA "289", "ARZU TANIŞ", 3, 5, 3, 4
410 DATA "396", "M.AKIN ACAR", 5, 5, 8, 8
420 DATA "428", "NEDİM KORKMAZ", 4, 6, 8, 4
430 DATA "218", "RIDVAN TANIŞ", 6, 7, 4, 9
440 DATA "365", "BELMA TANIŞ", 8, 4, 5, 7
450 DATA "563", "ALİ SEVİNÇ", 6, 7, 9, 9
460 DATA "376", "AYSEL ERKOL", 5, 4, 8, 9
1000 DATA "000", "AAAA", 0, 0, 0, 0
```

Bu program isteğe göre öğrencilerin tamamının ya da numarası verilen öğrencinin dökümünü yapar.

BÖLÜM XII. EK BİLGİLER

A.BASIC PROGRAMLAMA DİLİNDE EKRANA ÇIKABİLECEK HATA MESAJLARI

<u>Hata Adı:</u>	<u>Kodu:</u>	<u>Anlamı:</u>
* NEXT without FOR	1	NEXT'in FOR'u yok
* Syntax Error	2	Yazım hatası
* RETURN without GOSUB	3	RETURN'un GOSUB'u yok
* Out of data	4	Data'lar bitti. Data dışına taşma hatası
* Illegal function call	5	Geçersiz fonksiyon kullanımı
* Overflow	6	Taşma var
* Out of memory	7	Bellek dışına çıkma var
* Undefined line number	8	Tanımlanmamış satır numarası
* Subscript out of range	9	İndis tanımlanan değer dışına taşı
* Duplicate definition	10	Çift tanım
* Division by zero	11	Sıfıra bölme var
* Illegal direct	12	Geçersiz direct komutu
* Type mismatch	13	Tip uyumsuzluğu
* Out of string space	14	String boşluğu dışına taşma var
* String too long	15	String çok uzun
* String formula too complex	16	Formül çok karışık
* CAN't continue	17	Devam etmek olanaksız
* Undefined user function	18	Tanımlanmamış kullanıcı fonksiyonu
* No resume	19	RESUME yok
* Resume without error	20	Hata yok ama RESUME var
* Unprintable error	21	Yazılamayan, mesajı olmayan hata
* Missing operand	22	İfade eksikliği. Unutulmuş operand
* Line buffer overflow	23	BUFFER'de taşma var
* Device time out	24	Zamanaşımından aygıt devre dışı
* Device fault	25	Çevre ünitelerde aygıt hatası
* FOR without NEXT	26	FOR'un NEXT'i yok
* Out of paper	27	Yazıcıda kağıt yok
* While without wend	29	WHILE'in WEND'i yok
* WEND without WHILE	30	WEND'in WHILE'ı yok
* Field overflow	50	Alanda taşma var
* Internal error	51	İç hata
* Bad file number	52	Geçersiz dosya numarası

* File not found	53	Dosya bulunamadı
* Bad file mode	54	Geçersiz dosya modu
* File already open	55	Dosya halen açık
* Device I/O error	57	Aygıt giriş/çıkış hatası
* File already exist	58	Dosya zaten var
* Disk full	61	Disk dolu
* Input past end	62	Okuma sona erdi
* Bad record number	63	Geçersiz kayıt no
* Bad file name	64	Geçersiz dosya adı
* Direct statement in file	66	Dosyada direkt komut
* Too many files	67	Çok fazla dosya var
* Device unavailable	68	Çevre ünite uygun değil
* Disk write protect	70	Disk yazma için engelli, korumada
* Disk not ready	71	Disk hazır değil
* Disk media error	72	Diskte hata var

B. MS-DOS HATA MESAJLARI

Bu bölümde **MS-DOS** işletim sisteminde ekrana çıkabilecek hata mesajlarından, en çok karşılaşılabileceğimiz mesajları göreceğiz.

Abort Edit (Y/N)? :

Edlin kullanımı sırasında, Q komutu kullanıldığında MS-DOS tarafından çıkar. Edlinden çıkarken Q komutu kayıt yapmadan yani değişiklikleri kaydetmeden çıkar. Çıkmak istiyorsanız Y tuşuna, geri dönmek için N tuşuna basılır.

Abort, Retry, Fail?- :

Böyle bir mesaj çıktığında disk ya da aygıt hatası var demektir. Aşağıda belirtilen tuşlardan biri ile cevap verilir.

A Abort: Son verilen komutu iptal ederek eski konumuna döner.

R Retry: Son verilen emri yerine getirmesi için yeniden zorlar. Emri bir kez daha yerine getirmeye çalışır.

F Fail: Yeni bir bir komut verilmesi için eski komutu durdurur.

Access denied:

Yazım korunumlu (Write Portected), yalnızca okunabilir bir dosya üzerinde değişiklik yapılmaya kalkıldığını belirtir.

Are you sure (Y/N)? :

Bulduğumuz dizindeki bütün dosyaları *.* ile silmeye kalktığımızda bu mesaj çıkar. Silmek isteyip istemediğimizi bize bir kez daha sorarak bizi uyarır. Y girersek silme olayı gerçekleşir, N girersek vazgeçilir.

Bad command or file name :

Bir komutu doğru yazmadığımız ya da dosya adını yanlış yazdığımız zaman çıkar. Yanlış komut ya da yanlış dosya anlamındadır.

Bad or missing Command Interpreter:

Command.com dosyası bulunamıyor. Bu diskette bilgisayarı açamazsınız. Command.com'un kopyalanması gerekir.

Bad or missing <Dosya Adı> :

Bilgisayarı açış sırasında çıkar. Sistem dosyalarının yanlış belirtilmesinden kaynaklanır.

Batch file missing :

Aranılan batch dosya bulunamamıştır.

Cannot find System Files :

Sistem dosyalarının bulunmadığı bir disket ya da dizin kullanmaya çalışıyorsunuz.

Cannot load COMMAND, system halted :

MS-DOS açılış bulunamıyor. Boot bozulmuş sistemi yeniden kopyalayınız.

Compare another diskette (Y/N)? :

Diskcomp karşılaştırma işlemi yapıldıktan sonra çıkar. Başka karşılaştırma olup olmadığını sorar.

Compare O.K. :

Diskcomp sırasında çıkar. Disklerin aynı olduğunu belirtir.

Copy another diskette (Y/N)? :

Diskcopy kullanımı sonrasında çıkar. Kopyalama işleminin tamamlandığını, başka kopyalama olup olmadığını sorar.

Current date is mm-dd-yy :

Date komutu kullanıldığında çıkar. Doğru tarih girilip Enter'e basılır.

Current time is hh:mm:ss.cc :

Time komutu kullanıldığında çıkar. Doğru zaman girilip Enter'e basılır.

Data error reading drive x: :

İşletim sistemi tarafından disk okunamamaktadır. Genelde diskin bozuk olduğu durumlarda çıkar.

Delete (Y/N)? :

Kontrollü olarak silme işlemi yapılırken ortaya çıkar. Sileyim mi diye sormaktadır.

Disk full error writing to BACKUP Log file :

Disketin dolu olduğunu belirtir. Dosyayı yeniden başlatmak için herhangi bir tuşa ya da Control+Break tuşlarına basın.

Disk unsuitable for system disk :

Disketi sistemli formatlarken, sistemin yerleştirileceği alanların bozuk olduğunu belirtir. Bu disketi yalnızca bilgi kaydı için kullanabilirsiniz.

Duplicate file name or file found :

Olan bir ismi yeniden verdiniz. Ya da yeniden adlandırmak istediğiniz dosya bulunamadı.

Entry error :

Yazdığımız komut hatalı yeniden yazınız.

Error in .EXE file :

Çalıştırılmak istenen .EXE dosyası geçersiz durumdadır. Muhtemelen kodlama hatası vardır.

Error loading operating system :

Disk hatası nedeniyle işletim sistemi hard diskten yüklenemiyor. Sistem disketinizle açıp, yeniden sys.com ile sistem transferi yaparak hard diskinize sistem dosyalarını yükleyin.

Error reading directory :

Diskteki uygunsuz alanlar nedeniyle dizin okunamıyor

Error writing directory :

Diskteki uygunsuz alanlar nedeniyle dizine bilgi yazılamıyor.

Expanded Memory not available :

Sistemde genişletilmiş bellek kartının olmadığını belirtir.

File not found :

Dosya adında bir yanlışlık olduğunu belirtir.

Format another (Y/N)? :

Format işleminin tamamlandığını belirtir. Başka formatlanacak disket olup olmadığını sorar.

Format complete :

Formatlama işleminin tamamlandığını belirtmektedir.

Format failure :

Disketin formatlanmadığını belirtiyor. Bu mesajla birlikte nedeni de verilir.

General failure [writing/reading] error in drive x:

Okuma veya yazma yapılamıyor. Disketinizi yeniden formatlayın.

Incorrect DOS Version :

MS-DOS komutlarını kullanırken çıkar. İşletim sistemi versiyonunun farklı olduğunu belirtir.

Incorrect parameter :

Parametre hatası. Yanlış parametre verdiniz.

Insert system diskette in drive x

and strike any key when ready :

Sistemde aranan io.sys ve msdos.sys adlı gizli dosyalar bulunamıyor. Sistem transferi yapmanız gerekir ya da sistemli bir disketi sürücüye takmanız gerekir.

Insufficient disk space :

Disk dolu olduğundan işlem yapılamıyor.

Insufficient memory :

Bilgisayarınızın belleği bu iş için yetersiz kalıyor.

Invalid directory :

Böyle bir dizin yok, ya da yanlış dizin ismi.

Invalid drive in search path :

Sürücü mevcut değil.

Invalid drive or filename :

Geçersiz dosya adı ya da sürücü adı.

Invalid drive specification :

Yanlış sürücü ismi. Böyle bir sürücü yok.

Invalid parameter(s) :

Belirlenen anahtar(lar) yanlış ya da geçersiz.

Invalid path :

Dizin ya da dosya yolu yanlış.

Memory allocation error.

Cannot load MS-DOS, system halted :

MS-DOS'u yeniden çalıştırmayı deneyin. Yine çalışmazsa sistem diskinin bir kopyasını alın.

--More-- :

Devamını görmek için ara çubuğuna basın.

No paper error writing device dev :

Yazıcıda kağıt yok ya da yazıcı kapalı.

Non-system disk or disk error

Replace and strike any key when ready :

Sistem, diski tanıyamıyor ya da diskte hata var. Diski değiştirip herhangi bir tuşa basın.

Parameters not compatible :

Birlikte kullanılmayacak parametre kullandınız.

Press any key to continue :

İşlemi sürdürmek için herhangi bir tuşa basın.

Printer error :

Yazıcı kapalı ya da hazır değil.

Program too big to fit in memory :

Kullanacağınız program belleğe sığmıyor.

Reading source file(s)... :

XCOPY belirlenen dosya(ları) okumaktadır.

Strike a key when ready ... :

Hazır olunca herhangi bir tuşa basın.

Syntax error :

Yanlış yazım. Komutu doğru yazıp yazmadığınızı kontrol edin.

System transfered :

Format ya da sys sırasında sistem tansferinin yapıldığını belirtir.

Target diskette bad or incompatible :

Hedef disk bozuk ya da tanımlanamayan bir formata sahip.

Terminate batch job (Y/N)? :

Uzantısı .BAT olan dosyalar çalışırken CONTROL+C ya da CONTROL+BREAK kullanıldığında işlemi yarıda kesip kesmeyeceğinizi sorar.

Unexpected DOS error n :

Belirlenemeyen hata. Hata numarası verilir.

Write failure, diskette unusable :

Diske yazım yapılırken ilginç bir hata oluştu. Disketinizi kontrol edin ya da yeniden formatlayın.

Write protect error

Format terminated :

Formatlamaya çalışılan disk protect (Yazım Korunumlu).

Write protect error writing drive x:

X sürücüsündeki disket yazım korunumlu bilgi kaydedemezsiniz.

C. ASCII, HEX ve KARAKTER KODLARI LİSTESİ

Karakter karşılıkların ilk 32'si komut, diğerleri karakterdir.

Dec	Hex	Chr	Dec	Hex	Chr	Dec	Hex	Chr	Dec	Hex	Chr
000	00	NUL	033	21	!	065	41	A	097	61	a
001	01	SOH	034	22	"	066	42	B	098	62	b
002	02	STX	035	23	#	067	43	C	099	63	c
003	03	ETX	036	24	\$	068	44	D	100	64	d
004	04	EOT	037	25	%	069	45	E	101	65	e
005	05	ENQ	038	26	&	070	46	F	102	66	f
006	06	ACK	039	27	'	071	47	G	103	67	g
007	07	BEL	040	28	(072	48	H	104	68	h
008	08	BS	041	29)	073	49	I	105	69	i
009	09	HT	042	2A	*	074	4A	J	106	6A	j
010	0A	LF	043	2B	+	075	4B	K	107	6B	k
011	0B	VT	044	2C	,	076	4C	L	108	6C	l
012	0C	FF	045	2D	-	077	4D	M	109	6D	m
013	0D	CR	046	2E	.	078	4E	N	110	6E	n
014	0E	SO	047	2F	/	079	4F	O	111	6F	o
015	0F	SI	048	30	0	080	50	P	112	70	p
016	10	DLE	049	31	1	081	51	Q	113	71	q
017	11	DC1	050	32	2	082	52	R	114	72	r
018	12	DC2	051	33	3	083	53	S	115	73	s
019	13	DC3	052	34	4	084	54	T	116	74	t
020	14	DC4	053	35	5	085	55	U	117	75	u
021	15	NAK	054	36	6	086	56	V	118	76	v
022	16	SYN	055	37	7	087	57	W	119	77	w
023	17	ETB	056	38	8	088	58	X	120	78	x
024	18	CAN	057	39	9	089	59	Y	121	79	y
025	19	EM	058	3A	:	090	5A	Z	122	7A	z
026	1A	SUB	059	3B	;	091	5B	[123	7B	{
027	1B	ESC	060	3C	<	092	5C	\	124	7C	
028	1C	FS	061	3D	=	093	5D]	125	7D	}
029	1D	GS	062	3E	>	094	5E	^	126	7E	~
030	1E	RS	063	3F	?	095	5F	_	127	7F	□
031	1F	US	064	40	@	096	60	`	128	80	Ç
032	20	SPACE									

Dec	Hex	Chr	Dec	Hex	Chr	Dec	Hex	Chr	Dec	Hex	Chr
129	81	ü	161	A1	í	193	C1	⊥	225	E1	β
130	82	é	162	A2	ó	194	C2	⊥	226	E2	Γ
131	83	â	163	A3	ú	195	C3	⊥	227	E3	π
132	84	ä	164	A4	ñ	196	C4	—	228	E4	Σ
133	85	à	165	A5	Ñ	197	C5	⊥	229	E5	σ
134	86	â	166	A6	Ĝ	198	C6	⊥	230	E6	μ
135	87	ç	167	A7	ğ	199	C7	⊥	231	E7	τ
136	88	ê	168	A8	ı	200	C8	⊥	232	E8	φ
137	89	ë	169	A9	ı	201	C9	⊥	233	E9	⊙
138	8A	è	170	AA	ı	202	CA	⊥	234	EA	Ω
139	8B	ï	171	AB	½	203	CB	⊥	235	EB	δ
140	8C	î	172	AC	¼	204	CC	⊥	236	EC	∞
141	8D	ı	173	AD	i	205	CD	=	237	ED	φ
142	8E	Ä	174	AE	«	206	CE	⊥	238	EE	ε
143	8F	Å	175	AF	»	207	CF	⊥	239	EF	∩
144	90	É	176	B0	⊥	208	D0	⊥	240	F0	≡
145	91	æ	177	B1	⊥	209	D1	⊥	241	F1	±
146	92	Æ	178	B2	⊥	210	D2	⊥	242	F2	≥
147	93	ô	179	B3		211	D3	⊥	243	F3	≤
148	94	ö	180	B4	⊥	212	D4	⊥	244	F4	
149	95	ò	181	B5	⊥	213	D5	⊥	245	F5	
150	96	û	182	B6	⊥	214	D6	⊥	246	F6	÷
151	97	ù	183	B7	⊥	215	D7	⊥	247	F7	≈
152	98	ÿ	184	B8	⊥	216	D8	⊥	248	F8	°
153	99	Ö	185	B9	⊥	217	D9	⊥	249	F9	·
154	9A	Ü	186	BA	⊥	218	DA	⊥	250	FA	·
155	9B	Ç	187	BB	⊥	219	DB	■	251	FB	√
156	9C	£	188	BC	⊥	220	DC	■	252	FC	n
157	9D	¥	189	BD	⊥	221	DE	■	253	FD	z
158	9E	Ş	190	BE	⊥	222	DF	■	254	FE	■
159	9F	ş	191	BF	⊥	223	E0	■	255	FF	
160	A0	á	192	CO	L	224	E1	α			

BİLGİİŞLEM TERİMLERİ SÖZLÜĞÜ

access.....erişme	commandkomut
accessory.....aksesuar	comments.....açıklama
adaptor.....uyarlayıcı	communications.....iletişim
addition.....toplama	comparekarşılaştırma
adressing.....adresleme	compatibility.....uyarlık
analysis.....analiz	compatibleuyumlu
android.....erkek robot	compiler.....derleyici
animation.....animasyon	composite.....karma
annotation.....akış çizelgesi	computer centerbilgisayar merkezi
appendekleme	computer gamebilgisayar oyunu
application.....uygulama alanı	computer networkbilgisayar ağı
archivearşiv	computer systembilgisayar sistemi
array.....dizi	computer utility.....bilgisayar desteği
array processor.....dizi işlemci	computer wordbilgisayar sözcüğü
assembler.....çevirici	condition.....koşul
availability.....kullanılrlık	connecting cablekablo bağlantısı
binary.....ikili	console.....konsol
binary digitikili sayı	continuedevam
breakara	control key.....kontrol tuşu
brush.....firça	control unitdenetim birimi
buffer.....tampon	conventiongelenek
cablekablo	conversiondönüştürme
cache memory.....ön bellek	converter.....çevirici
cage.....kafes	coordinatekoordinat
calculatorhesap makinesi	copykopyalama
calibration.....ayar	countersayaç
callçağırma, hücre	current.....akım
cancel.....iptal	cursor.....imleç, ışıklı gösterge
capbüyük harf	cycle.....dönüş
cardkart	cylinder.....silindir
cardridgekartuş	dataveri
centrall/processing unit (CPU) işlem birimi	data baseveri tabanı
channelkanal	data processingbilgi işlem
characterkarakter	decimalonluk sayı
chipçip, yonga	decode.....kod çözme
clearingsilme	deletesilme
clocksaat	densityyoğunluk
closed.....kapalı	device.....aygıt
codekod	diagram.....diyagram
collatorbirleştirici	dibit.....ikili bit
colorrenk	dictionary.....sözlük
column.....sütun, kolon	digit.....sayı
	digital cominationdijital iletişim

digital computer.....	sayısal bilgisayar	flat.....	bayrak
dimension	boyut	flow.....	akış
direct access.....	direkt erişim	flow chart.....	akış şeması
direct current.....	direkt akım	font	yazı karakteri
directory.....	dizin	form	biçim
disk access time	disk erişim süresi	format	biçim
disk copying.....	disk kopyalama	formula	formül
disk file	disk dosyası	function.....	fonksiyon
display	görüntüleme	fuse	sigorta
documantion	dökümantasyon	generator	üretici
document	belge	graph.....	grafik
dot matrix printer.....	nokta matris yazıcı	grounding.....	topraklama
double density.....	çifte yoğunluk	group.....	grup
double-sided disk.....	çift yüzlü disk	gunched card.....	delikli kart
drain.....	çizim	gynoid	dişi robot
drive.....	sürücü	hardware	donanım
dynamic	dinamik	hybrid.....	melez
echo	eko	initalize	başlatmak
edit.....	biçimlendirmek	input.....	girdi
editör.....	biçimlendirici	insert	araya sokma, yerleştirme
effective	etkili	install	kurma
element	eleman	integer	tamsayı
embeddet	gömülü	inter face	ara birim, ara yüz
emulator	uyarlayıcı, benzetici	interactive	etkileşim
end-of-file.....	dosya sonu	interface	ara birim
entry.....	giriş	internal clock	iç saat
environment.....	çevre	interpreter	yorumlayıcı
equality	eşitlik	interrup	kesme
equation	denklem	invert.....	geri alma, ters çevirme
erasable storage.....	silinebilir bellek	job.....	iş
erase.....	silme	joystick	kumanda çubuğu
error	hata	jump.....	atlama
escape key.....	kurtulma tuşu	key	anahtar, tuş
execution	işletim	keyboard	klavye
exit.....	çıkış	keyword	anahtar sözcük
expression.....	ifade	kill.....	silme, çıkarma
face	yüz	kilobit	bin bit
factor.....	faktör	kilosycle.....	bin devir
fatal error	ölümcül hata	language.....	dil
fault.....	hata	language processor	dil işlemci
feed	besleme	leader	klavuz
feedback.....	geri besleme	level	düzey
file.....	dosya	library	kitaplık
file organization.....	dosya düzenleme	line	hat, çizgi, satır
file structure.....	dosya yapısı	line printer	satır yazıcı
file transfer.....	dosya aktarma	link.....	bağlantı
filling.....	doldurma	list.....	liste

list structure	liste yapısı	optimum.....	en uygun
literal.....	sabit	or.....	veya
load.....	yükleme	output.....	çıkıtı
local.....	yemel	output area	çıkıtı alanı
location.....	bölge	output device	çıkıtı aygıtı
lock.....	kilit	overflow.....	taşma
logarithm	logaritma	pack	sıkıştırmak
logic.....	mantık	package.....	paket
loop.....	döngü	package software.....	paket yazılım
macro	makro	page	sayfa
magnetic disk.....	manyetik disk	page printer.....	sayfa yazıcı
main memory.....	ana bellek	parallel printer	paralel yazıcı
marker.....	işaretleyici	parameter.....	parametre
matrix.....	matris	parity.....	eşlik
matrix printer.....	matris yazıcı	parrallel conversion	paralel erişim
medium.....	ortam	password.....	parola
memory.....	bellek	path.....	yol
memory chip.....	bellek çipi	perform.....	uygulamak
memory map.....	bellek haritası	performance.....	performans
message.....	mesaj	personal computing.....	kisisel bilgisayar
microchip.....	mikroçip	pixel.....	eleman, nokta
microcomputer.....	mikrobilgisayar	pocket computer	cep bilgisayarı
microprocessor	mikroişlemci	point.....	nokta
module.....	birim	port.....	kapı
monitor	ekran	portable computer.....	taşınabilir bilgisayar
morge.....	birleştirmek	power.....	güç
mother board.....	ana levha	power supply	güç kaynağı
mouse.....	fare	press.....	basamak
mouse button	fare düğmesi	primitive.....	ilkel
move.....	taşıma	print head.....	baskı kafası
multi programming.....	çoklu programlama	print quality	baskı kalitesi
multiplex.....	çok düzeyli	printer.....	yazıcı
multiplier	çarpan	processor.....	işlemci
multiply.....	çarpma işlemi	programmable.....	programlanabilir bellek
network.....	ağ	programmer	programlamacı
object computer.....	amaç bilgisayar	programming	programlama
object program.....	amaç program	protect.....	koruma
off-line.....	çevrim dışı	prototype.....	prototip
on-line.....	çevrim içi	push.....	itme
open.....	açma	quality.....	kalite
operand.....	bilgi	radian.....	radyan
operating system.....	işletim sistemi	random access.....	rastgele erişim
operator.....	işleç	read head.....	okuma kafası
optical character reader.....	optik karakter okuyucu	record.....	kayıt
optical disk.....	optik disk	reduction.....	kısaltma
optical page reader.....	optik sayfa okuyucu	relation.....	ilişki
optical scanner.....	optik tarayıcı	relative.....	göreceli

reliability	güvenirlilik	structute	yapı
remainder	kalan	sub program	alt program
repeat	tekrar	subdirectory	alt dizin
report	rapor	supercomputer	süperbilgisayar
reproduce	çoğaltmak	switch	anahtar
resident program	yerleşik program	syntax	yazım, söz dizim
rigger	tetikleme	sysntax error	yazım hatası
run	yürütmek	system	sistem
save	kaydetme	system analyst	sistem analisti
scan	tarama	system programmer	sistem programlayıcısı
scanner	tarayıcı	system software	sistem yazılımı
screen	ekran	table	tablo
search	arama	tape	teyp
second	saniye	target	hedef
security	güvenlik	target disk	hedef disk
selection	seçim	tele communications	tele komünikasyon
sequantial access	sıralı erişim	template	kalıp
sequence	sıra	testing	deneme
serial	seri	text	metin
set	takım	three	ağaç, dallı yapı
setup	düzenek	tone	ton
shade	gölge	trace	izleme
shift	kaydırma	track	iz
silicon chip	silikon çip	transfer	aktarmak
simlex	tek yönlü	transform	dönüştürmek
simulation	benzetim, gösterim	user	kullanıcı
single density	tek yoğunluk	utility	yardım, hizmet
skip	atlama	value	değer
slot	yarık	variable name	değişken isim
soft copy	geçici kopya	version	uyarlama
software	yazılım	vertical	dikey
sort	sıralama, düzenleme	warning message	uyarı mesajı
source	kaynak	warranty	garanti
source disk	kaynak disk	window	pancere
space	boşluk	word	sözcük
specification	belirleme	word processing	kelime işlemci
speed	hız		
stack	yığın, küme		
start bit	başlangıç biti		
state	durum		
statement	deyim, ifade		
statistic	istatistik		
step	basamak		
storage	bellek		
store	saklama		
string	dizgi		
stroke	vuruş		

İNDEKS

BİLGİSAYARA GİRİŞ ve MS-DOS

A	Conventional Memory . 11	DRIVER.SYS 133
Alt.....42	COPY 68	E
ALT DİZİN 32	C.P.U 10	ECHO..... 124
Alt Gr42	CPS..... 17	EDIT 114
ANA DİZİN32	Ctrl..... 41	EGA 15
Analog Bilgisayar.....4	Ç	Eklenebilir Bellek 11
ANSI.SYS 132	Çizici 25	EKRAN..... 13
ATTRIB 89	D	EMM386.EXE 134
AUTOEXEC.BAT 123	DATE 55	EMS 11
B	DBLSPACE..... 101	End..... 46
Back Space.....43	DBLSPACE.SYS 132	Enter..... 41
BACKUP 84	DEBUG 121	EPROM..... 12
BATCH..... 122	DEFRAG 102	ERASE..... 76
BELLEK 10	DEL 76	Esc..... 43
Bilgi..... 1	Del 46	EXPAND 119
Bilgi İşlem..... 1	Delete..... 46	Expanded Memory 11
Bilgisayar 1	DELTREE 77	Extended Memory..... 11
Bit.....5	DEVICE 129	F
BREAK 128	DEVICEHIGH 130	Fare 23
BUFFERS 128	DIR 59	FDISK..... 104
Byte.....5	DIRECTORY 32	FILE 34
C	DISKCOMP 90	FILES..... 130
CALL 125	DISKCOPY 74	FIND 108
Caps Lock.....43	DISPLAY.SYS 133	Floppy 18
CD 66	Dış Komut..... 38	FOR.IN.DO 125
CD Disk.....21	Digital Bilgisayar..... 4	FORMAT 50
CGA 15	Disket..... 18	Formatlama 18; 50
CHDIR 66	Dizin 32	G
CHKDSK 79	DOS 130	Genişleyebilir Bellek ... 11
CHOICE..... 126	DOSKEY 82	Giga Byte 6
CLS 65	Dosya..... 34	GOTO 125
COMP 89	Donanım 6	GRAFTABL 111
CONFIG.SYS..... 128	Dot Matrix Printer 16	GRAPHICS..... 112
COUNTRY 129	dpi..... 17	
	DRIVE..... 22	

H	M	Print Screen..... 44
Hard Disk21	MD..... 65	PRINTER.SYS 134
HARDWARE.....6	Mega Byte 6	PROM 12
HIMEM.SYS..... 134	MEM 97	PROMPT 96
High Memory Area 11	MEMMAKER 98	R
HMA 11	MENUCOLOR..... 131	RAM BELLEK 11
Home46	M.İ.B. 10	RAMDRIVE.SYS 135
Hybrid5	mili second..... 21	RD 67
I	MKDIR..... 65	REM..... 127; 132
IF125	MODE 112	REN 79
Ink-Jet Printer.....17	Monitör 13	RENAME..... 79
Insert.....45	Monochrome..... 13	REPLACE..... 74
Işık Kalemı25	MOVE 73	RESTORE..... 85
i	MSAV..... 92	RETURN 41
İç Komut.....38	MSBACKUP 86	RMDIR 67
İşletim Sistemi 8; 30	MSD 119	ROM BELLEK 12
J	MS-DOS 30	ROOT DIRECTORY ... 32
Joystick.....25	Mürekkep Püskürtmeli Yazıcı..... 17	S
K	N	Sabit Disk..... 21
KEYB..... 108	Network 25	Satır Yazıcılar 17
Kilo Byte.....6	NLQ..... 17	SCANDISK..... 81
Klavye 15; 39	Nokta Vuruşlu Yazıcı 16	Scanner..... 23
Kullanılabilir Bellek 11	Num Lock 47	Scroll Lock..... 45
KÜTÜK.....34	NUMLOCK..... 131	SECTOR 18
L	O	SETVER 107
LABEL.....57	Ok tuşları 45	SETVER.EXE..... 135
LASTDRIVE..... 131	Oyun Çubuğu..... 25	Shift..... 41
Lazer Yazıcılar 17	P	SHIFT 127
Light Pen25	Page Down..... 46	SOFTWARE 8
LH 100	Page Up 46	SMARTDRV 106
LOADFIX 100	PATH..... 32; 92	SMARDRV.SYS..... 136
LOADHIGH..... 100	Pause..... 45	SORT 110
LQ 17	PAUSE 127	Space..... 43
	Pixel..... 13	SUB DIRECTORY 32
	Plotter 25	Sürtücü 22
	Printer 16	SYS 58

T	Ü	X
Tab44	Üst Bellek 11	XCOPY 71
Tarayıcı23	V	XMS 11
Tera Byte.....6	VER 57	Y
Termal Yazıcılar.....17	Veri..... 1	YAZILIM..... 8
Teyp25	VERIFY..... 132	Yazıcı..... 16
TIME.....56	VGA 15	Yüksek Bellek 11
TRACK.....18	VOL..... 58	
TREE.....63	VSAFE 91	
TYPE.....78		
U	W	
UNDELETE.....93	WINDOWS-NT 30	
UNFORMAT95		
UNIX.....30		
UMA11		
Upper Memory Area11		

PCTOOLS

A		M	
Attrib	154	Map.....	174
B		Move.....	147
Beginning Menü.....	139	P	
C		Park.....	185
cOmp	148	Paste.....	159
cOmpare	169	Print	160
COPY	145	R	
Copy	159; 167	Rename	151; 171; 180
create	181	remove	181
Cut	159	Replace	159
D		S	
Delete	151	Save	158
Directory maint	179	Scroll Lock OFF	141
E		Scroll Lock ON.....	142
ENTER.....	142	Search	158
Esc.....	143	Select	159
exit.....	143	Sort	164
F		U	
Find	148; 171	Undelete.....	181
H		UNselect	142
Help.....	165; 185	V	
I		Ver	152
Info	183	Verify.....	172
iNitalize.....	177	view/Edit.....	173
L		W	
List	163	wiew/Edit.....	153
Locate.....	176	Wordp.....	156

PW

A

Add an address	232
Add word to dictionary.....	231
Address.....	232
Ana Menü.....	189

B

Bloklama	188
Bold.....	198
Boldface word	219
Bottom margin.....	225

C

Calculate.....	218
Calcute	222
Cansel.....	188
Center	227
Center line	229
Change a font	209
Change Data Directory	192
Change font or size.....	218
Change indentation.....	218
Change left/right margins (Ctrl-[]).....	218
Change margins of the current block	224
Change paper size.....	209
Change Screen Colors	193
Change Screen Update Speed.....	194
Change style	217
Change tabs (Ctrl-K)	218
Change tabs of the current block	226
Change view.....	208
Change Work Drive	193
Comment	232
Compressed	206
Continue	188
Continue without replacing	222
Copy (Ctrl-C)	217

Create / Edit	189
Ctrl PrtSc-Preview	206
Cut.....	188; 217

D

Data file to merge.....	206
Del.....	217
Delete file.....	204
Delete line	215
Delete word	215
Description.....	203
Document or envelope (D/E)	206
Double space.....	218
Draw lines	220
Drawing.....	198
Durum Cetveli.....	195

E

Edit.....	188
Editör	187
End and save recorded keystrokes	213
Ending page	205
Erase a macro.....	213
Erase working copy.....	211
Exit.....	189; 194

F

F1 HELP	195
F1-Help	207
F2 File/Print	200
F2-Options	207
F3 Edit	214
F3-Fonts	209
F4-Format	223
F4-Return	210
F5-Dictionary	230
F6/Addresses.....	231
File type	203

PROGRAMLAMAYA GİRİŞ ve BASIC

A	COLOR..... 362; 366	FIX..... 345
ABS..... 347	COMMON..... 368; 369	FOR..NEXT..... 332
ADA..... 264	CONT..... 274; 289	FORTRAN..... 263
Akış şeması 255	COS..... 346	FRE..... 373
Alfasayısal Değişkenler. 262	CSNG..... 348	G
Alfasayısal Sabitler..... 261	CSRLIN..... 355	GET #..... 387; 389
Algoritma 251	Ctrl..... 279	GET (Grafik)..... 363
ALL..... 368	CVD..... 390	GOSUB-RETURN..... 324
Alt..... 277	CVI..... 390	GOTO..... 316
Analist..... 250	CVS..... 390	H
AND-OR-NOT..... 321	D	HEX\$..... 354
Append..... 377	DATA..... 376	HOME..... 276
ASC..... 354	DATE\$..... 356	I
ATN..... 347	Değişkenler..... 262	IF-THEN-ELSE..... 319
AUTO..... 315	DEF..... 371	INKEY\$..... 355
B	DEF FN..... 371	INPUT..... 300
BASIC..... 264	DEL..... 277	INPUT #..... 379
BEEP..... 315	DELETE..... 282	INPUT#..... 376
C	DIM..... 340	INPUT\$..... 355
C..... 264	Direkt Mod..... 280	Input)..... 377
CDBL..... 347	DRAW..... 360	Insert..... 277
CHAIN..... 368	E	INSTR..... 353
CHDIR..... 370	EDIT..... 313	INT..... 345
CHR\$..... 349	END..... 288	K
Ç	End..... 276	Katar..... 261
Çift Duyarlıklılı Sayı. 269	Endirekt Mod..... 280	KEY..... 275; 374
C	EOF..... 376; 379	KILL..... 286
CINT..... 345	ERASE..... 344	L
CIRCLE..... 359	ERL..... 372	LEFT\$..... 350
CLEAR..... 344	ERR..... 372	LEN..... 350
CLOSE..... 391	Escape..... 279	LET..... 297
CLOSE #..... 376; 378; 387	EXIT..... 316	LINE..... 358
CLS..... 287	EXP..... 348	LINE INPUT..... 356
COBOL..... 263	F	LINE INPUT #..... 376; 380
	FIELD..... 387; 388	
	FILES..... 285	

LIST	272; 282	POKE.....	373	SOUND.....	365		
LLIST.....	272; 282	POS.....	355	SPACE\$.....	351		
LOAD.....	274; 284	PRESET.....	362	SQR.....	346		
LOC.....	376; 380; 387	PRINT	290	STEP.....	333; 335		
LOCATE.....	309; 367	PRINT #.....	376; 378	STOP.....	289		
LOF.....	376; 380; 387	PRINT USING	310	STR\$.....	352; 390		
LOG.....	346	Program.....	250	String.....	261		
LOGO.....	264	Programcı.....	250	STRING\$.....	349		
LPRINT.....	295	Programlama.....	250	SWAP	352		
LSET.....	390	PSET.....	361	SYSTEM.....	286		
M			T				
MERGE.....	368	PUT #.....	387; 389	TAB.....	308		
MID\$.....	350	PUT (Grafik).....	363	Tab.....	278		
MKD\$.....	390	R					
MKDIR.....	370	RANDOM	387	Tam Sayı.....	268		
MKIS.....	390	RANDOMIZE	348	TAN.....	347		
MKS\$.....	390	RANDOMIZE TIMER.....	349	Tek Duyarlıklık Sayı.....	269		
N			READ-DATA.....	304	TIMES\$.....	356	
NAME ... AS.....	285	REM.....	288	TROFF.....	275		
NEW.....	283	RENUM.....	314	TRON.....	275		
NumLock:.....	278	RESET.....	369	Ü			
O			RESTORE.....	307	Üslü Sayılar.....	261	
OCT\$.....	354	RESUME.....	372	V			
ON ERROR GOTO.....	372	RIGHT\$.....	351	VAL.....	351; 390		
ON KEY.....	375	RMDIR.....	370	W			
ON..GOSUB	326	RND.....	348	WHILE NOT..WEND.....	339		
ON..GOTO.....	326	RPG.....	264	WHILE..WEND.....	339		
OPEN ... 376; 377; 387; 388		RSET.....	390	WIDTH.....	357; 365		
OPTION BASE.....	340	RUN.....	273; 285; 288	WIEW.....	357		
Output.....	377	S				WIEW PRINT	358
P			Sabitler.....	261	WINDOW.....	358	
PAINT.....	363	SAVE.....	273; 283	WRITE.....	297	WRITE #.....	376; 379
PASCAL	264	SAYAÇ.....	329	'			
PEEK.....	373	Sayısal Değişkenler.....	262	,"LPT1:".....	275		
PL/1.....	264	Sayısal Sabitler.....	261	,			
PLAY.....	364	SCREEN.....	357	,			
POINT.....	363	SCREEN 0,0,0.....	276	,			
		SGN.....	348	,			
		SHELL.....	315	,			
		SIN.....	346	,			

KAYNAKÇA

- BİLGİSAYAR ANSİKLOPEDİSİ, Milliyet Yayınları. Milliyet Tesisleri, İstanbul 1991
BİLGİSAYAR PAZARI Dergisi, Ankara
BİLİM ve TEKNİK DERGİSİ, TÜBİTAK Yayınları, Ankara
BOLU TİCARET MESLEK LİSESİ, Hizmet İçi Eğitim Kursu Ders Notları
DOS, Korkmaz, Halim. Ankara 1993
DOS Power User's Guide.
GW-BASIC Made Easy.
GW-BASIC Reference.
İLERİ BASIC ve GWBASIC, Y.Doç.Dr. Mazmanoğlu, Adnan. İstanbul, Haziran 1989
MICROSOFT GW-BASIC Interpreter, 1986
MICROSOFT MS - DOS, 1986
PC HELP Dergisi, Ankara
PC TOOLS DELUXE MADE EASY, Greg M. Perry, McGraw-Hill
PROFESSIONAL WRITE AND FILE MADE EASY, Bryan Pfaffenberger, McGraw-Hill