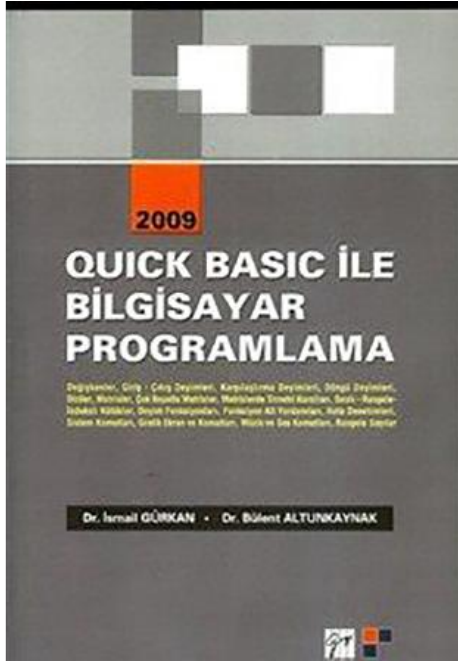




SAB104 Bilgisayar Programlama

Hafta 1 – Programlamanın Tanımı

Prof.Dr. Fatih TANK
Ankara Üniversitesi
Uygulamalı Bilimler Fakültesi



SAB104 Bilgisayar Programlama

dersine ait sunumlar hazırlanırken ağırlıklı olarak

Quick Basic ile Bilgisayar Programlama

(Dr. İsmail Gürkan, Dr. Bülent Altunkaynak)

kitabından yararlanılmıştır





GİRİŞ

Bütün programlama dillerine bakıldığında aşağıdaki özellikler gözlenir.

- Dilin alfabesi
- Değişmezler
 - Sayısal değişmezler
 - Alfa sayısal değişmezler
 - Mantıksal değişmezler
- Değişkenler
 - Sayısal değişkenler
 - Alfa sayısal değişkenler
 - Mantıksal değişkenler
- Aritmetik ifadeler
- Okuma deyimi
 - Ekrandan okuma
 - Diskten okuma
- Manyetik şeritten okuma
- Yazma deyimi
 - Ekran yazma
 - Diske yazma
 - Manyetik şeride yazma
 - Satır yazıcıya yazma
- Karşılaştırma (Koşul) deyimi
- Döngü yapıları
- Dizinli değişkenler
- Dizgi değişkenler
- Hazır fonksiyonlar
- Deyim fonksiyonları
- Alt programlar
- Kütükler (dosyalar) ile ilgili deyimler



- Bütün programlama dillerinin alfabesi genel olarak aynıdır.
 - A ile Z arasındaki İngiliz harfleri,
 - 0 ile 9 arasındaki rakamlar
 - Özel işaretler
 - Genel olarak klavye üzerine gördüğümüz ve yukarıda sayılmayan karakterlerdir.
 - Bunlar dışında, klavyede görülmeyen bir takım karakterler de kullanılmaktadır.
- QBasic'te, programda yer alan her deyimin tamsayı değer olan bir deyim numarası bulunmalıdır.
- Genel işleyiş olarak programda yer alan deyimler, deyim numarası küçükten büyüğe doğru sıralanmalıdır.
- Program, aksi söylenmediği sürece, deyimlerin en küçük deyim numarasından başlanarak, sırayla, deyimlerin işlenmesi yoluyla yürütülür.

100	██████████
200	██████████
300	██████████
400	██████████
500	██████████
600	██████████
700	██████████

- QBasic' te bir satıra iki nokta (:) ile ayrılmış birden çok deyim yazılabilir.

```
200 K=0 : T=5 : M= N*N
300 IF T=0 THEN MM=N/2 : ORT=TOPLAM/N
```



- Açıklama satırı: İşlem görmez sadece açıklama yapar:

```
(1)  REM -----
      REM  Karekök bulma programı
      REM  -----
           CLS
           K=0

(2)  \-----
      \  Newton-Raphson yöntemi
      \  -----
           DEF FNX (x)=2*X*X+3*X-1
           DEF FNT (x)=2*X+3
```

- Değişmezler

150%
1654356&
9288333722#
1672345!
"CADDE"

Tamsayı
Uzun tamsayı
Çift duyarlıklı
Tek duyarlıklı
Dizgi

```
CONST PI = 3.141593
INPUT "Dairenin yarıçapı = "; r
PRINT "Alan = "; PI * r ^ 2
```

- Tür Bildirileri

INTEGER	16-bit işaretli tamsayı,
LONG	32-bit işaretli tamsayı,
SINGLE	32-bit tek duyarlıklı, kayan noktalı,
DOUBLE	64-bit çift duyarlıklı, kayan noktalı,
STRING	değişken uzunlukta dizgi,
STRING*n%	n% uzunlukta,sabit uzunlukta dizgi

değişkeni belirler.

```

REM -----
REM Tür belirtme hataları için örnek program
REM -----
CLS
DIM K, N AS INTEGER
DIM M, L, S AS DOUBLE
DIM D, E, F AS SINGLE
L = 65.924: M = 76.23
a# = 23.75: b# = 48.72
D = 15.65: E = 45.96
S = L * M
K = L * M
c% = a# * b#

D# = a# * b#
F = D * E
N = D * E
PRINT "L (çift d)                =", L
PRINT "M (çift d)                =", M
PRINT "Beklediğiniz sonuç (L*M)  =", S
PRINT "Alacağınız sonuç (L*M)    =", K
PRINT "a# (çift d)                =", a#
PRINT "b# (çift d)                =", b#
PRINT "Beklediğiniz sonuç (a#*b#) =", D#
PRINT "Alacağınız sonuç (a#*b#)  =", c%
PRINT "65.924*76.23 doğrudan çarpımı =", 65.924 * 76.23
PRINT "D (tek d)                 =", D
PRINT "E (tek d)                 =", E
PRINT "Beklediğiniz sonuç (D*E)  =", F
PRINT "Alacağınız sonuç (D*E)    =", N
END

```



- Aritmetik işlemler ve deyimler

Aritmetik işlemler ve öncelikleri

İşleç	Öncelik
^	1
Tekil -	2
*, /	3
+, -	4

Cebirsel ifade

$$A^2+B^2$$

$$\frac{X}{Y} + \frac{C+D}{2} + \frac{D-A}{X}$$

$$T1 - \frac{T1^2}{N}$$

$$N - 1$$

QBasic karşılığı

$$A^2+B^2$$

$$X/Y+(C+D)/2+(D-A)/X$$

$$(T1-T1^2/N)/(N-1)$$

(parantez kullanımına dikkat ediniz)

- Mantıksal işlemler ve deyimler

NOT	Değildir
AND	Ve
OR	Veya
XOR	Yalnız
EQV	Denklik
IMP	Gerektirme

İşleç	Öncelik sırası
NOT	En yüksek
*, /, MOD, AND	↓
+, -, OR	
<, <=, =, <>, >=, >	

Mantıksal bağlantı elemanları

Bağlantı Elemanı	Anlamı	İfade
=	Eşittir	A = B
<>, > <	Eşit değildir	A <> B
<	Küçüktür	A < B
>	Büyüktür	A > B
>=	Büyük, eşittir	A >= B
<=	Küçük, eşittir	A <= B

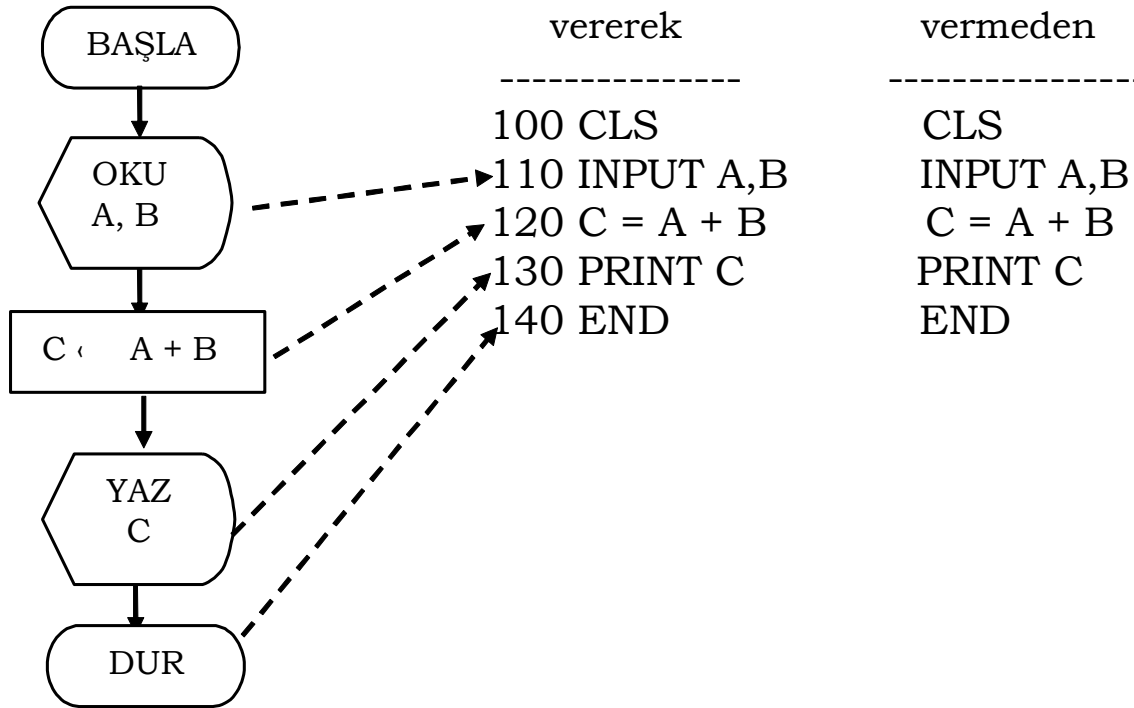
- Aşağıdaki mantıksal ifadelerde A, B, ve C gerçel sayılardır. A=6.0, B=8.0 ve C=4.0

(A>C) AND (B>C)	Doğru
(A+B/C) <= 7.0	Yanlış
(C>A) OR (C>B)	Yanlış
(A=2.0) OR (A=6.0)	Doğru
(C<A) AND (A<B)	Doğru
(A<=C) OR (A>=B)	Yanlış



- Ekrandan okunan iki sayının toplamını bulup yazdıran QBasic programı.

Akış Şeması :



Eğer, akış şemasını (doğru) çizebiliyorsak, kodlama zor bir iş değildir

BAŞLA - DUR

EKRANA
YAZMA

VERİ GİRİŞİ / Klavye

İŞLEM

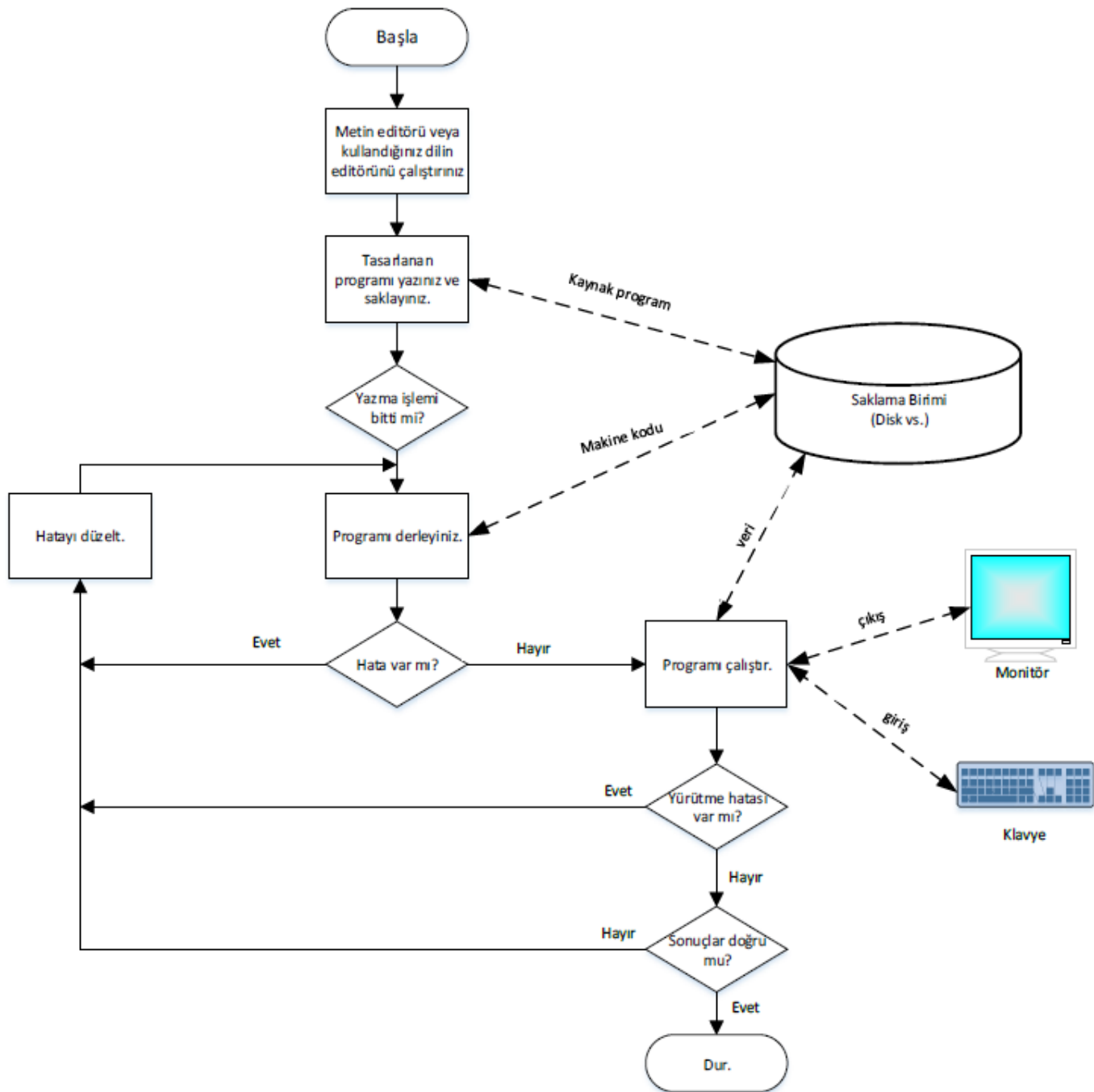
DÖNGÜ

FONKSİYON
SUBROUTINE

DISKTEN
OKUMA

YAZICI ÇIKTISI

İKİLİ
KARAR





GİRİŞ-ÇIKIŞ DEYİMLERİ

- **CLS:** Ekranı tümünü siler. Genel yazılışı, CLS [{0 | 1 | 2}]
 - CLS 0 :Bütün yazı ve grafik ekranını temizler.
 - CLS 1 :Sadece grafik materyalleri siler veya grafik konumunda değil ise bütün ekranı temizler
 - CLS 2 : Yazı materyallerini siler.

- **LOCATE:** imleci ekranın belirlenen konumuna hareket ettirir.

LOCATE 10,10 : PRINT "10. satır 10. kolon"

LOCATE SAT,SUT

- **CSRLIN:** imlecin o anda bulunduğu pozisyonu verir.

```
CLS
LOCATE 5, 5, 1
Satir% = CSRLIN
Sutun% = POS(0)
PRINT "000000000111111111222222222222"
PRINT "123456789012345678901234567890"
PRINT "*" ; Satir% ; Sutun%
DO
LOOP WHILE INKEY$ = ""
LOCATE (Satir% + 4), (Sutun% + 2), 1
PRINT "*" ; Satir% ; Sutun%
END
```

- **INPUT:** Ekrandan veri okutmak için kullanılır

```
200 INPUT A,B,C
400 INPUT DEGER1,DEGER2
300 INPUT "Bir sayı giriniz...>",SAYI
160 INPUT "Virgülle ayırarak 4 değer giriniz
...>",DEG1,DEG2,DEG3,DEG4
500 INPUT "", Değişken listesi
```



- **INKEY\$:** Klavyeden sadece bir (1) karakter okur

```
CLS
KONTROL = 1
WHILE KONTROL = 1
LOCATE 10, 10
K$ = INKEY$
IF K$ = CHR$(27) THEN
    KONTROL = 0
ELSE
    IF K$ = "A" THEN PRINT "A ...ya bastınız"
    IF K$ = "B" THEN PRINT "B ...ya bastınız"
    IF K$ = "C" THEN PRINT "C ...ya bastınız"
LOCATE 10, 15
PRINT K$; " ...tuşuna bastınız"
END IF
WEND
END
```

- **LINE INPUT:** bir kütük veya klavyeden en çok 255 karaktere kadar olan bir satır okur.

```
LINE INPUT "Tam adres giriniz";ADRES$
LINE INPUT ; "Ad soyad giriniz"; ADSOY$
```

- **READ...DATA:**

```
250 READ A,B,C,D
...
...
600 DATA 3,5,3
610 DATA 9
```

A	B	C	D
3	5	3	9

- **TAB:** Yazı imlecini belirlenen yazma pozisyonuna hareket ettirir.
- **GOTO:** Programın ilerleyişi sırasında, programın akış yönünü değiştirmek amacıyla kullanılır.



- **RESTORE:** READ deyimi ile DATA deyim (leri) ardına sıralanan veri değerlerinin yeniden (baştan başlayarak) okunması amacı ile kullanılır.
- **PRINT:** Programda elde edilen sonuçları, açıklamaları ekrana yazdırmak için kullanılır.

```
200 PRINT A,B,C
400 PRINT "İterasyon sonucu = ...",ITER
410 PRINT "İterasyon sonucu = ...";ITER
300 PRINT "A matrisi elemanları "
160 PRINT USING " ###.##   ###.##";SAYI1,SAYI2
```





KARŞILAŞTIRMA DEYİMLERİ

- **IF THEN ELSE ENDIF:** Bir problemin çözümü sırasında, bir koşulun sonucuna göre yapılacak işlemler farklılık gösterebilir. Diğer bir deyişle, koşul doğru olduğunda yapılacaklar ile, yanlış olduğunda yapılacaklar birbirinden farklı olabilir. Bu işlem (karşılaştırma) QBasic'te IF deyimini ile gerçekleştirilir.

```
(1) IF Koşul THEN deyim(ler)
(2) IF Koşul THEN deyim(ler) ELSE deyim(ler)
(3) IF Koşul THEN
    Deyim bloğu-1
    ELSE
    Deyim bloğu-2
    ENDIF
```

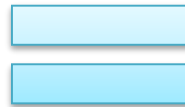
```
REM -----
REM Bu program ekrandan girilen üç sayının en büyüğünü
REM bulup yazdırır
REM -----
INPUT "Virgül ile ayırarak üç sayı
giriniz...>",A,B,C
IF A > B THEN
    IF A > C THEN
        PRINT "A en büyük"
    ELSE
        PRINT "C en büyük"
    ENDIF
ELSE
    IF B > C THEN
        PRINT "B en büyük"
    ELSE
        PRINT "C en büyük"
    ENDIF
ENDIF
END
```

```
REM -----
REM Bu program ikinci derece denklem köklerini bulur.
REM -----
INPUT "Virgül ile ayırarak ikinci derece
denklemin katsayılarını giriniz.....>",A,B,C
DELTA = B*B-4*A*C
IF DELTA > 0 THEN
    X1 = (-B + SQR(DELTA))/(2*A)
    X2 = (-B - SQR(DELTA))/(2*A)
    PRINT "1. Kök   =";X1
    PRINT "2. Kök   =";X2
ELSE
    IF DELTA = 0 THEN
        X1 = -B / (2*A)
        PRINT "Çift Kök   =";X1
    ELSE
        PRINT "Gerçek kök yok"
    ENDIF
ENDIF
END
```



- **ON ... GOTO:** Koşullu sapma deyimi

```
CLS
INPUT "DEĞER GİRİNİZ...>", DEGER
IF DEGER = 1 THEN GOTO BIR
IF DEGER = 2 THEN GOTO IKI
IF DEGER = 3 THEN GOTO UC
IF DEGER = 4 THEN GOTO DORT
GOTO YANLISGIRIS
BIR:
  A = SQR(DEGER)
  GOTO YAZ
IKI:
  A = DEGER * DEGER
  GOTO YAZ
UC:
  A = DEGER ^ 3
  GOTO YAZ
DORT:
  A = DEGER ^ 4
  GOTO YAZ
YANLISGIRIS:
  PRINT "Hatalı değer girdiniz"
  GOTO SON
YAZ:
  PRINT A
SON:
  END
```



```
CLS
INPUT "DEĞER GİRİNİZ...>", DEGER
ON DEGER GOTO BIR, IKI, UC, DORT
GOTO YANLISGIRIS
BIR:
  A = SQR(DEGER)
  GOTO YAZ
IKI:
  A = DEGER * DEGER
  GOTO YAZ
UC:
  A = DEGER ^ 3
  GOTO YAZ
DORT:
  A = DEGER ^ 4
  GOTO YAZ
YANLISGIRIS:
  PRINT "Yanlış değer girdiniz"
  GOTO SON
YAZ:
  PRINT A
SON:
  END
```



- **CASE:** Gizli karşılaştırma yapısı

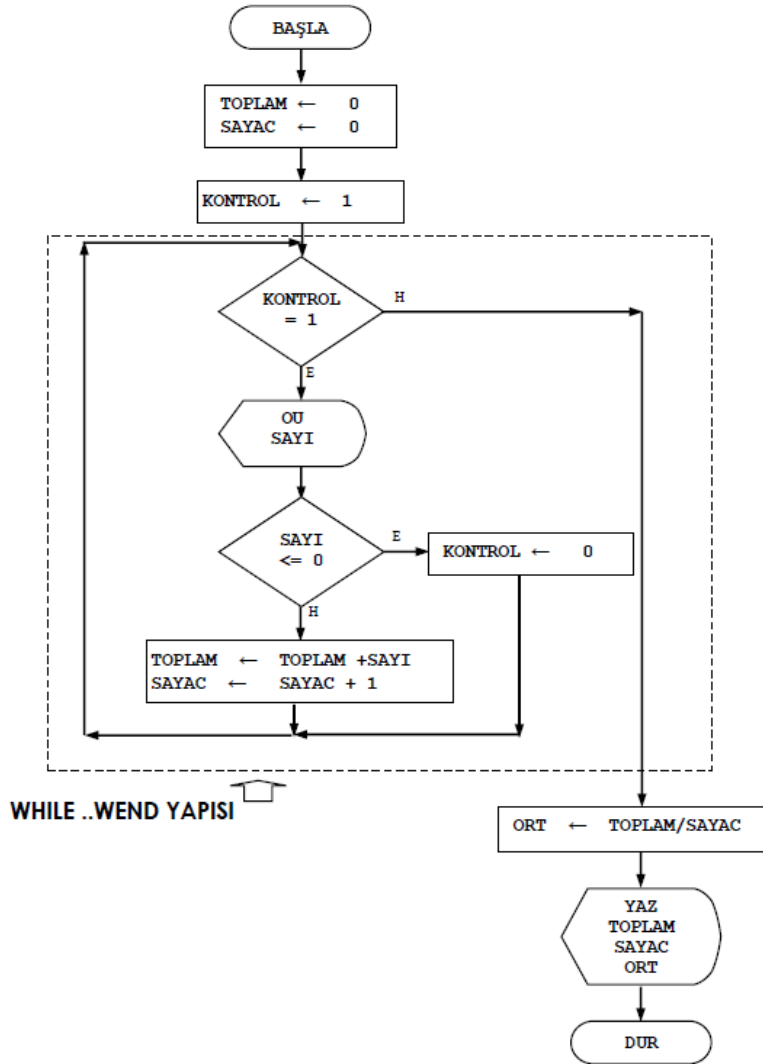
```
-----  
Case yapısı örnek (1)  
-----  
CLS  
INPUT "DEĞER GİRİNİZ...>", DEGER  
SELECT CASE DEGER  
  CASE 1  
    K = SQR(DEGER)  
    PRINT K  
  CASE 2  
    K = DEGER * DEGER  
    PRINT K  
  CASE 3  
    K = DEGER ^ 3  
    PRINT K  
  CASE 4  
    K = DEGER ^ 4  
    PRINT K  
  CASE ELSE  
    PRINT "Yanlış değer girdiniz"  
END SELECT  
END
```





YİNELEME DEYİMLERİ

- **WHILE...WEND:** Genel olarak, tekrar edilecek, fakat tekrar sayısı bilinmeyen deyim veya deyim gruplarının tekrarlanmasını sağlayan bir yapıdır.



CLS

```

TOPLAM = 0
SAYAC = 0
KONTROL = 1
WHILE KONTROL = 1
    INPUT "Sayı giriniz...>", SAYI
    IF SAYI <= 0 THEN
        KONTROL = 0
    ELSE
        TOPLAM = TOPLAM + SAYI
        SAYAC = SAYAC + 1
    END IF
WEND
ORT = TOPLAM / SAYAC
PRINT "Sayıların toplamı = "; TOPLAM
PRINT "Kaç sayı olduğu = "; SAYAC
PRINT "Sayıların ortalaması = "; ORT
END
  
```

- **DO..LOOP:** Bir deyimler bloğunu bir koşul doğru iken veya doğru olana kadar tekrarlar.

```
DO [{WHILE | UNTIL} koşul]
    [deyim bloğu]
LOOP
```

Veya

```
DO
    [deyim bloğu]
LOOP [{WHILE | UNTIL} koşul]
```

- **FOR ... NEXT:** Yinelenen ve yinelenme sayısı bilinen deyim veya deyimler grubu, FOR ve NEXT deyimlerin arasına konulmak yoluyla, daha kısa ve anl aşılabılır bir yapıda tasarlanabilir.

```
FOR I = 1 TO 10
    FOR J = 1 TO 10
        FOR K = 1 TO 10
            . . .
            . . .
        NEXT K
    NEXT J
NEXT I
```

```
FOR I = 1 TO 10
    FOR J = 1 TO 10
        FOR K = 1 TO 10
            . . .
            . . .
        NEXT J
    NEXT K
NEXT I
```





HAZIR MATEMATİKSEL FONKSİYONLAR

- **ABS**: bir sayının mutlak deęerini verir.
- **SGN**, verilen sayısal ifadenin iřaretini elde etmek için kullanılır.
- **ATN**, belirlenen sayısal ifadenin arctanjant'ını verir.
- **TAN**, verilen açı deęerinin tanjantını hesaplar.
- **COS**, verilen açı deęerinin kosinüsünü verir.
- **EXP**, e^x sayısının nümerik deęerini hesaplar.
- **LOG**, bir sayısal ifadenin doęal logaritmasını verir.
- **SIN**, verilen bir açı deęerinin sinüsünü bulur.
- **SQR**, verilen bir sayısal ifadenin karekökünü bulur.
- **INT**, sayısal bir ifadenin deęerinden küçük veya eřit olan en büyük tamsayıyı verir.
- **MODE**, İki sayısı bölerek, bölme iřleminin kalanını geri döndürür. (Modüler aritmetik).
- **CINT**, sayısal ifade deęerini tamsayıya yuvarlar.

