

Temel Bilgisayar Programlama ve BASIC Diline Giriş

Ders Notları

Selahattin ERGEÇ-Mustafa Y.ATA

KAPSAM

1. Temel Bilgisayar Donanım-Yazılım Bilgileri
2. Simgeler, Değişmezler ve Değişkenler
3. Aritmetiksel ve Mantıksal İşlemler
4. Algoritma ve İşlem Akış Çizgesi
5. Temel Girdi-Çıktı Deyimleri
 - 5.1 Girdi Deyimleri
 - 5.2 Çıktı Deyimleri
6. Yönlendirme Deyimleri
7. Döngüler
8. Matematiksel İşlevler
9. Dizgisel İşlevler

1. Temel Bilgisayar Donanım -Yazılım Bilgileri

- **Donanım:** Bilgisayarın bedeni.
 - Merkezi İşlem Birimi
 - Aritmetik ve Mantıksal İşlem Birimi
 - Girdi-Çıktı Birimi
 - Ana Bellek
 - Değişmez Bellek
 - Değişir Bellek
 - Çevre Birimleri
 - Ekran, Klavye,Fare,Yazıcı-Çizici-Tarayıcı,Hoparlör,Mikrofon,Kamera
 - Dış Bellek Sürücüleri
- **Yazılım:** Bilgisayarın ruhu.Bilgisayarın anlayacağı bir dille yapacağı işlerin tanımı.

Elektrik akımı ile çalışan bilgisayarın *doğrudan* anlayacağı dil, sözcükleri yalnızca “0-1” simgeleri kullanılarak yazılan “Makine Dili”dir.

Makine dilini kullanarak, bilgisayarın yapabileceği her işi tanımlamak olası değildir. Bu zorluk, “kullanımı kolay diller”le yazılmış yönergeleri aşama-aşama Makine Diline çevirecek yazılımlarla aşılar. Bir bilgisayar programlama dilinin kullanımı kolaylaştıkça üst düzey bir programlama dili olma niteliği ağırlık kazanır.

BASIC, FORTRAN, C gibi diller üst düzey diller arasında yer alır. Bu dillerin sözcüklerini, *programın çalışması sırasında adım adım* makine diline çeviren yazılıma, söz konusu programlama dilinin “çeviricisi-yorumlayıcı” denir.

- İşletim Yazılımları
- Uygulama Yazılımları.

2. Simgeler,Değişmezler ve Değişkenler

- Simgeler
 - Alfabetik simgeler, A-z
 - Sayısal simgeler 0-9
 - Özel simgeler + - , ; . * ? = () / & % ! vb.
- Değişmezler: Bellekte belli bir adresteki değişmeyen değerlerdir.
 - Alfasayısal Değişmezler: Üzerinde aritmetik işlem yapılmayan değişmezlerdir. Alfasayısal değişmezlerin değerleri genelde “ “ içinde gösterilir. “Merhaba”, “Ad-Soyad.”, “A2212210” gibi.

- **Sayısal Değişmezler:** Üzerinde aritmetik işlem yapılabilen değişmezlerdir.
 - Tamsayı değişmezler
 - Gerçel değişmezler
- **Değişkenler:** Program süresince değişik değerler yerleştirilebilen bellek adreslerinin simgesel adlardır.
 - **Alfasayısal Değişkenler:** Alfasayısal değişkenlerin bellek adresleridir. Adres adının sonunda \$ simgesi yer alır. A\$, B45\$, AdSoyad\$ gibi.
 - **Sayısal Değişkenler:** Sayısal değerlerin bellekteki adres adlarıdır.
 - Tamsayı değişkenler: Adres adının sonunda % simgesi yer alır. A%, B45% vb.
 - Gerçel değişkenler: A, B45 vb.

Kural: Değişken adı, sayısal ya da özel simgelerden biri ile başlayamaz.
A, B4, C\$ değişken adı olarak doğru; 4B, \$C ise yanlıştır.

3. Aritmetiksel ve Mantıksal İşlemler

- **Aritmetik İşlemler:** Sayısal değişmez ve değişkenlerin bellek adreslerindeki değerlerinden “aritmetik işlemler” ile “mantıksal işlemler” den denklik işlemleri kullanılarak tanımlanan “cebirsal deyimler”le yeni değişmez ve değişken değerleri üretir.

– İşlem öncelik sırasına göre Aritmetik İşlemler:

- Parantez: ()
- Üs: ^
- Bölme ve Çarpma: / *
- Toplama ve Çıkarma: + -

– Mantıksal İşlemler:

- Denk : =
- Büyük : >
- Büyük veya denk : >=
- Küçük : <
- Küçük veya denk : <=
- Denk Değil : <>
- Veya : OR
- Ve : AND

– Cebirsal Deyimler:

A=5 ve b=3 ise, $C=2*3+(6-A)^b$ cebirsal deyimini ile, bellekteki A ve b sayısal değişmezlerinin değerleri, cebirsal ifadedeki yerlerine konup aritmetik işlem öncelik sırası gözetilerek

$$\begin{aligned}
 &2*3+(6-5)^3 \\
 &2*3+1^3 \\
 &2*3+1 \\
 &6+1 \\
 &7
 \end{aligned}$$

biçiminde adım adım elde edilen 7 değeri, sayısal bir değişken değeri olarak C adresine yerleştirilir.

- **Mantıksal İşlemler:** Mantıksal işlemlerle tanımlanan basit veya karmaşık mantıksal deyimlerin doğru ya da yanlış olduğuna karar verir. Bu mantıksal karar sonucuna göre işlem akışına yön verilir.

4. Algoritma ve İşlem Akış Çizgesi

Algoritma:

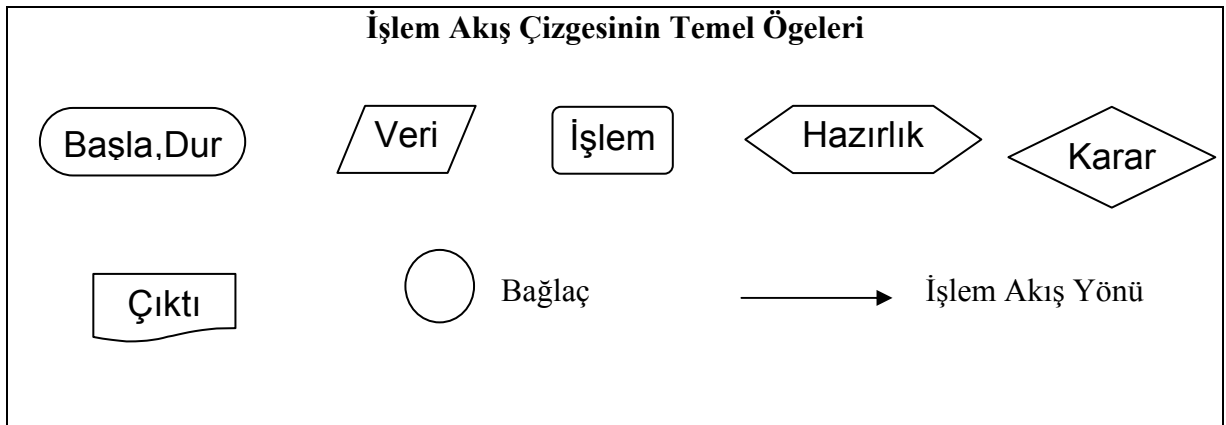
- Tüm ön bilgileri,
- Her adımda yapılacak işlemleri ve
- Sonucun istendiği biçimi

ile bilgisayarın yerine getirmesi beklenen işin mantıksal akış düzeni içinde tanımlandığı bir bilgisayar programının temel biçimidir.

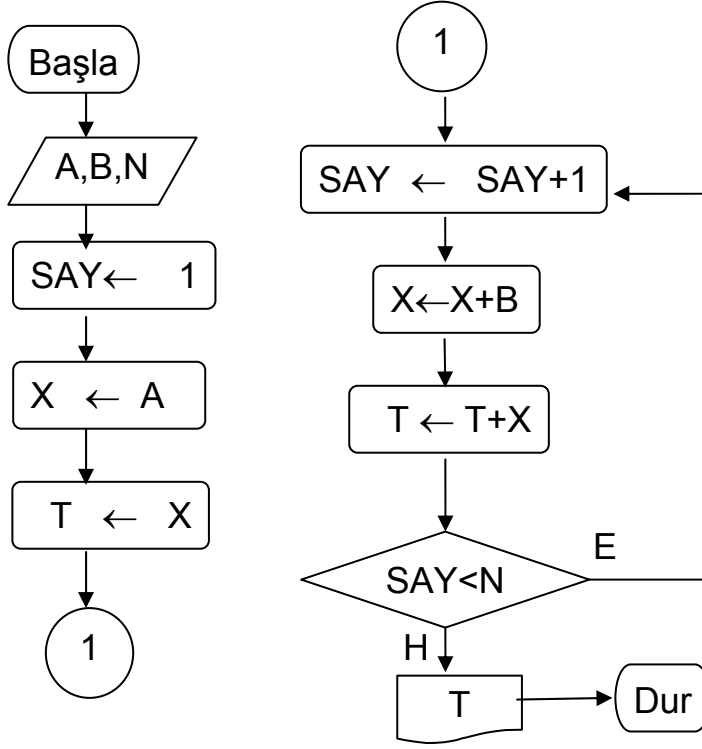
N terimli $T=a+(a+b)+((a+b)+b)+\dots\dots\dots$ toplamının algoritması:

- Adım 1: A, B, N değerlerini al.
 Adım 2: SAY \leftarrow 1
 Adım 3: X \leftarrow A
 Adım 4: T \leftarrow X
 Adım 5: SAY \leftarrow SAY +1
 Adım 6: X \leftarrow X+B
 Adım 7: T \leftarrow T+X
 Adım 8: Eğer SAY<N ise, 5'e dön.
 Adım 9: T değerini yaz.

İşlem Akış Çizgesi: Algoritması verilen bir bilgisayar programının başlatılması ve durdurulmasını da içeren ve her hangi bir dilde kodlanmasına olanak tanıyan işlem akışının çizgesel biçimi.

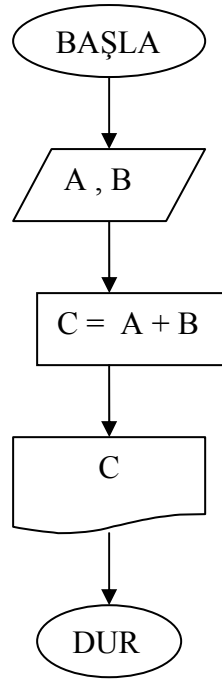


Yukarıda verilen örnek algoritmaya göre oluşturulan işlem akış çizgesi:



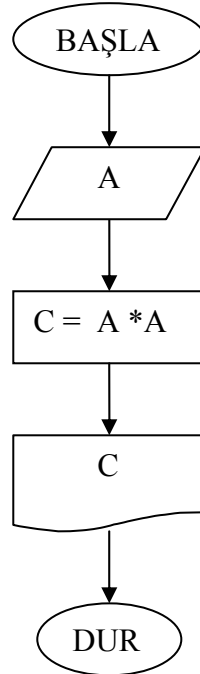
Örnek 1: İki sayının toplamı.

- Adım:1 Başla.
- Adım:2 A,B'yı gir.
- Adım:3 $C=A+B$
- Adım:4 C'yi yaz.
- Adım:5 Dur.



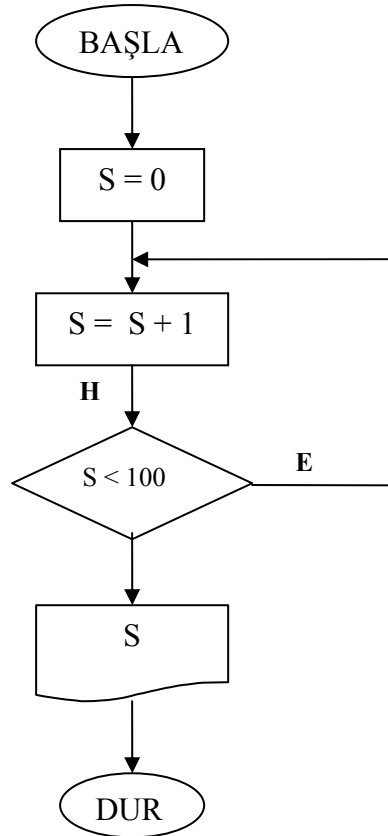
Örnek:2 Bir sayının karesi.

- Adım:1 Başla.
- Adım:2 A'yı gir.
- Adım:3 $C=A*A$
- Adım:4 C'yi yaz.
- Adım:5 Dur.



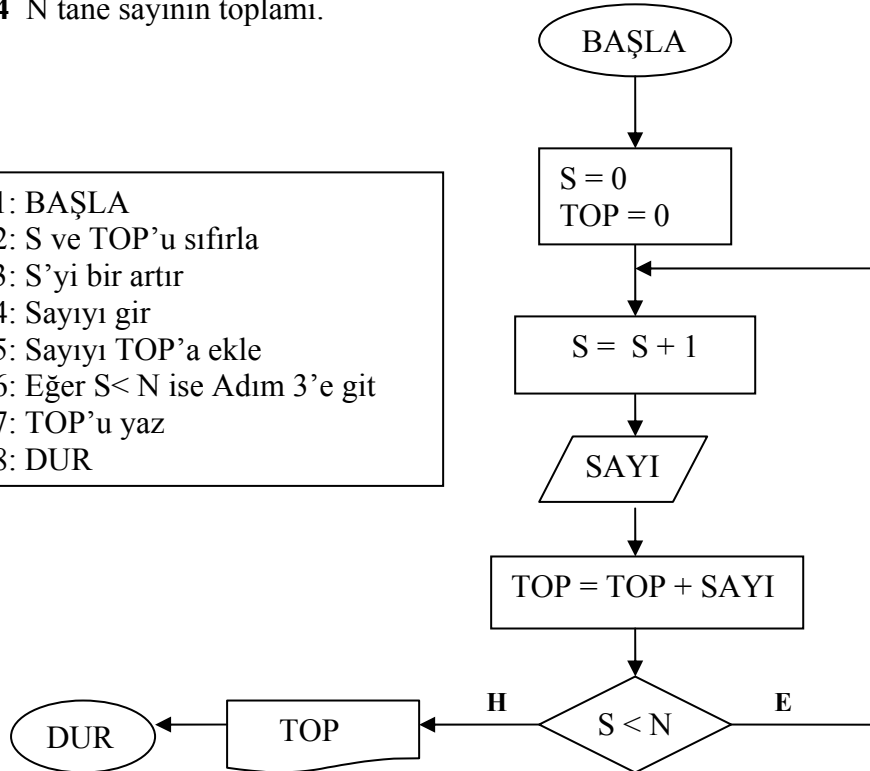
Örnek: 3 1 den 100' e kadar sayıların toplamı.

Adım:1 Başla.
 Adım:2 $S=0$
 Adım:3 $S=S+1$
 Adım:4 $S < 100$ ise Adım:3'e dön
 Adım:5 S'yi yaz.
 Adım:6 Dur.



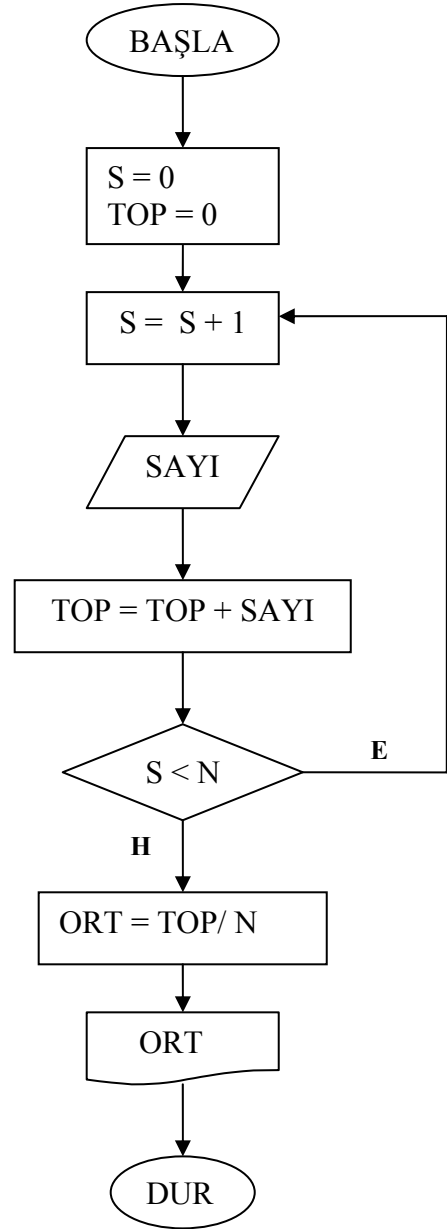
Örnek: 4 N tane sayının toplamı.

Adım 1: BAŞLA
 Adım 2: S ve TOP'u sıfırla
 Adım 3: S'yi bir artır
 Adım 4: Sayıyı gir
 Adım 5: Sayıyı TOP'a ekle
 Adım 6: Eğer $S < N$ ise Adım 3'e git
 Adım 7: TOP'u yaz
 Adım 8: DUR



Örnek: 5 N tane sayının ortalaması.

Adım 1: BAŞLA
Adım 2: S ve TOP'u sıfırla
Adım 3: S'yi bir artır
Adım 4: SAYI'yı gir
Adım 5: SAYI'yı TOP'a ekle
Adım 6: Eğer $S < N$ ise Adım 3'e git
Adım 7: $ORT = TOP / N$ hesapla
Adım 8: ORT'u yaz
Adım 9: DUR



Örnek 6: N öğrenciden oluşan bir sınıfta, notu 50'nin üstünde olan öğrencilerin sayısını belirleyen programın algoritması ve akış çizelgesi.

Açıklama: Burada iki sayaca gereksinim var. Biri genel sayaç (S), diğeri ise notu 50'nin üstünde olan öğrencilerin sayısını belirleyen sayaç (T)

Adım 1: BAŞLA

Adım 2: N'yi gir

Adım 3: S ve T'yi sıfırla

Adım 4: S'yi bir artır

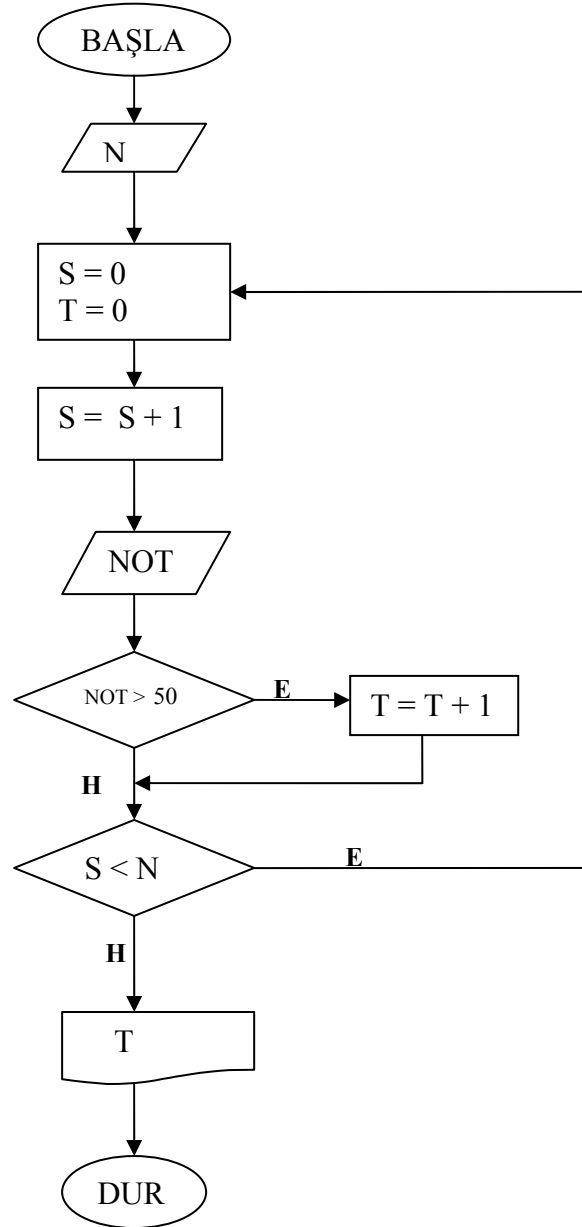
Adım 5: Notu gir

Adım 6 Eğer Not>50 ise T'yi 1 arttır.

Adım:7 Eğer S<N ise Adım:4'e git

Adım:8 T'yi yazdır.

Adım:9 DUR



5. Temel Girdi-Çıktı Deyimleri



- **Girdi** bileşeninde, değişken ve değişmezlerin değerleri bellekteki adreslere yerleştirilir.
- **İşlem** Bileşeninde, girilen girdi değerleri alınarak, tanımlanan algoritmaya göre aritmetik ve mantıksal işlemler yapılır.
- **Çıktı** bileşeninde, bellekteki adreslerde bulunan en son değerlere göre işlem sonuçları alınır.

Bilgisayarla iletişim yolu olan bir bilgisayar programlama dilinin de kendine özgü bir yapısı ve bir sözlüğü bulunur. Söz konusu bilgisayar dilinin yapısına uygun biçimde olmak üzere bu sözlükte yer alan sözcüklerle kurulan ve bir istem bildiren tümceler, o dilin deyimlerini oluşturur.

5.1 Girdi Deyimleri

INPUT[“Açıklama”]; DeğişkenAdı1, DeğişkenAdı2,.....

INPUT deyimi, program çalışırken, kullanıcı tarafından bir değişkene değer verilmesini sağlar.

Program akışı içinde,

```

      INPUT A
      deyim ile, ekranda
      ?_
  
```

simgesinin sağında yanıp sönen imleçle, kullanıcıdan A değişkeni için klavyeden bir değer girilmesi istenir ve kullanıcı her hangi sayısal bir değer girdiğinde, girilen değer A değişkeninin bellekteki adresine yazılması sağlanır.

LET

LET deyimi ile, bir değişkenin alacağı değer tanımlanır. Tanım, sayısal bir değer ya da daha önce tanımlanmış değişkenleri de içeren cebirsel bir işlem olabileceği gibi, değişken alfasayısal ise, tanım alfasayısal bir değer ya da ileride açıklanacak olan *alfasayısal işlemlerle* gerçekleştirilebilecek bir işlem de olabilir. Tanım sonucu türü ile değişken türü aynı olmalıdır.

[LET]DeğişkenAdı=Tanım

Program akışı içinde,

```
A$="TEMEL" : A=5 : B=A^2+3
```

deyimleri, bellekte A\$ adresine alfasayısal "TEMEL" değerini, A adresine sayısal 5 değerini, ve B adresine denkleğin sağındaki aritmetik işlem sonucu olan 28 değerinin yazılmasını sağlar.

READ - DATA

- READ deyiminde yer alan virgülle ayrılmış değişkenlere, DATA deyimindeki virgülle ayrılmış sabit değerler bire-bir biçiminde atanır.

```
READ DeğişkenAdı1, DeğişkenAdı2,.....
```

```
DATA SabitDeğer1, SabitDeğer2,.....
```

Program akışı içinde,

```
READ A$, A
DATA "TEMEL", 5
```

deyimleri, bellekte A\$ adresine alfasayısal "TEMEL" değerini ve A adresine de sayısal 5 değerinin yazılmasını sağlar.

- DATA deyimini ile başlayan satır, programın her hangi bir satırında bulunabilir.
- READ satırındaki değişken adı sayısı en az , DATA satırlarındaki sabit değer sayısı kadar olmalıdır.

RESTORE [SatırNumarası]

Bir BASIC programında istenen satırlara numara verilebilir. Tekrar okunması istenen değerleri içeren DATA satırına numara verilerek, tekrar okunacak değerler belirlenebilir.

Program akışı içinde,

```
READ B$, A$
DATA "DİL"
10 DATA "ANA"
DATA "SÜT"
RESTORE 10
READ C$, D$
```

deyimleri, bellekte B\$ adresine "DİL" değerini ve A\$ adresine "ANA"; C\$ adresine tekrar "ANA", ve D\$ adresine de "SÜT" alfasayısal değerlerinin yazılmasını sağlar. RESTORE deyimini ile birlikte bir satır numarası verilmese idi, C\$ ve D\$ adreslerine sırası ile, "DİL" ve "ANA" alfasayısal değerleri yazılacaktı.

- RESTORE deyimini ile, DATA satırından READ deyimini ile okunarak tüketilen değerlerin tekrar okunabilmesi sağlanır.

5.2 Çıktı Deyimleri

PRINT , LPRINT

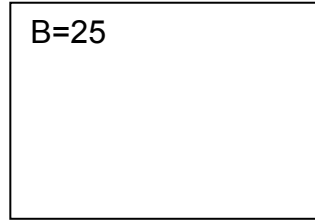
PRINT ve LPRINT deyimleri ile bellekteki adreslerden değişmez ve sabitlerin en son değerlerinin ekrana ya da yazıcıya istenen düzende yazdırılması sağlanır.

PRINT [“Açıklama” ;] DeğişkenAdı

Program akışı içinde,

```
A=5:B=A^2
PRINT “B=”;B
```

deyimleri, bellekteki B adresinden sayısal 25 değerini alarak “B=” açıklamasından hemen sonra ekrana



biçiminde yazılmasını sağlar. PRINT yerine **LPRINT** kullanılması durumunda, aynı sonucun yazıcıdaki kağıda yazdırılması sağlanır.

LOCATE

LOCATE deyimi ile ekrana yazdırılacak bilginin, istenen satır ve sütundan başlayarak yazdırılması sağlanır.

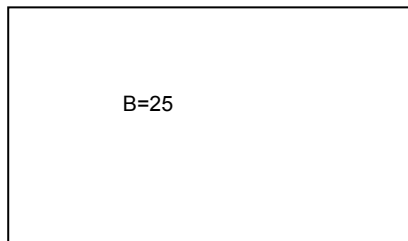
LOCATE SatırNo,SütunNo

PRINT deyiminden önce LOCATE deyimi kullanılarak, imleç istenen satır ve sütuna getirilir. Ekranda, 25 satır ve 80 sütun bulunur. Satır ve sütun numaraları bu değerleri aşmamalıdır.

Program akışı içinde,

```
A=5:B=A^2
LOCATE 10,25
PRINT “B=”;B
```

deyimleri, ekranın 10. satırına 25. sütunundan başlayarak



yazılmasını sağlar.

TAB(SütunSıraNo), SPC(BoşlukSayısı)

TAB() ve SPC() deyimleri PRINT ve LPRINT deyimleri ile birlikte kullanılır ve ekrana yazdırılacak bilginin, istenen sütundan başlayarak yazdırılmasını sağlar.

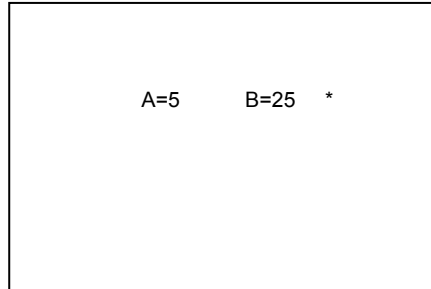
TAB (SütunNo); yazdırma sırasında imleci istenen sütuna getirir

SPC(Sütun Sayısı); en son yazılan simgeden sonra imleci Sütun Sayısı kadar ileriye taşır.

Program akışı içinde,

```
A=5:B=A^2
LOCATE 10,25
PRINT "A=";A;TAB(10); "B=";B;SPC(5);"*"
```

deyimleri, ekranın 10. satırına 25. sütunundan başlayarak



A=5 B=25 *

yazılmasını sağlar.

6. Yönlendirme Deyimleri

Bir BASIC programında istenen satırlara numara verilerek kontrol deyimleri ile program akışının belli bir satıra yönlendirilebilir. İşlem akışı kontrol deyimleri ile yönlendirilmedikçe, BASIC programı satır sırasını izler. Yönlendirme koşulsuz ve koşullu olabilir.

6.1 Koşulsuz Yönlendirme

GOTO SatırNumarası

İşlem akışının satır numarası verilen bir BASIC satırına yönlendirilmesi için, koşulsuz GOTO deyimini kullanılır.

Program akışı içinde,

```
10 INPUT A
   B = A^2
   PRINT B
   GOTO 10
```

deyimleri, kullanıcının klavyeden gireceği sayısal değeri bellekte A adresine yazar, A'nın karesini alarak sonucu B adresine yazar, B adresindeki değeri ekrana çıktı olarak verir ve kullanıcıdan yeni bir A değeri girmesini ister.

6. 2 Tek Koşullu Yönlendirme

ON İfade GOTO Satır Numaraları Listesi

İşlem akışının satır numarası verilen bir BASIC satırına koşullu olarak yönlendirilmesi için, koşullu GOTO deyimini kullanılır.

Program akışı içinde,

Program:6.2.1

```

10 INPUT A
   ON A GOTO 20,30,40,50
20 PRINT "BİR"
30 PRINT "İKİ"
40 PRINT "ÜÇ"
   GOTO 10
50 END

```

deyimleri, kullanıcı klavyeden A adresine "1" değerini girerse, 20 numaralı satırdaki işlemi; "2" değerini girerse, 30 numaralı satırdaki işlemi; "3" değerini girerse, 40 numaralı satırdaki işlemi yapar ve 10 numaralı satırdaki işleme döner. Kullanıcı "4" değerini verirse program akışı sona erer.

IF Mantıksal İfade THEN Eylemsel İfade [ELSE Eylemsel İfade]

Mantıksal ifade *doğru ise* "THEN" sözcüğünden sonraki eylemsel ifadenin gereği yerine getirilir. Mantıksal ifade *yanlış ise*, "ELSE" sözcüğünden sonraki eylemsel ifadenin gereği yapılır; "ELSE" sözcüğü kullanılmamışsa, program akışı doğrudan bir sonraki satıra geçer.

Algoritması ve işlem akış çizgesi daha önce verilen örnek programın aşağıdaki BASIC kodunda yer alan 1 numaralı satır, SAY değeri kullanıcı tarafından klavyeden girilen N değerine eşit olana kadar, program akışının yönünü 5 numaralı satıra çevirerek, toplamı alınacak rakamların adım adım sıra numaralarını SAY; değerlerinin X; ve toplamlarının da T adresine yazılmasını ve SAY=N olduğunda, dolayısı ile SAY < N ifadesi yanlış olduğunda, işlem akışının 3 numaralı satıra geçmesini sağlar.

Program:6.2.2

```
INPUT A, B, N
SAY = 1 : X = A : T = X
5 SAY = SAY + 1
  X = X + B : T = T + X
1 IF SAY < N THEN 5
3 PRINT T
END
```

Yukarıdaki programda, 1 ve 3 numaralı satırlar yerine yazılacak

```
IF SAY < N THEN 5 ELSE PRINT T
```

program satırının da aynı işlevi yerine getireceğine dikkat ediniz.

6.3 Çok Koşullu Yönlendirme

IF 1. Mantıksal İfade THEN

1. Eylemsel İfade

ELSEIF 2. Mantıksal İfade THEN

2. Eylemsel İfade

ELSEIF 3. Mantıksal İfade THEN

3. Eylemsel İfade

-
-
-

ELSE

Eylemsel İfade

ENDIF

1. Mantıksal ifade *doğru ise* “THEN” sözcüğünden sonraki 1. eylemsel ifadenin; 2. Mantıksal ifade *doğru ise* “THEN” sözcüğünden sonraki 2. eylemsel ifadenin; vd gereği yerine getirilir. Mantıksal ifadelerin hiç biri diğerleri ile örtüşmemelidir. Mantıksal ifadelerin

16-19 *Temel Programlama ve BASIC Diline Giriş* Selahattin ERGEÇ-Mustafa Y.ATA
hepsi de *yanlış ise*, “ELSE” sözcüğünden sonraki eylemsel ifadenin gereği yapılır; “ELSE” sözcüğü kullanılmamışsa, program akışı doğrudan bir sonraki satıra geçer.

Program akışı içinde

```
Program:6.3.1      5 INPUT “Negatif veya Pozitif Bir Sayı Giriniz.”; A
                   IF A = 0 THEN
                       GOTO 5
                   ELSEIF A>0 THEN
                       B = 100 / A
                   ELSE
                       B = (-1) * 100 / A
                   ENDIF
                   PRINT B
                   END
```

deyimleri, A değeri sıfır olarak girilmişse, tekrar A değeri almak için 5 nolu satıra; yok eğer A değeri pozitif bir değer olarak girilmişse, B değerini, 100’ün girilen A değerine oranı olarak; girilen A değeri ne sıfır ne de pozitifse, yani negatifse, B değerini, 100’ün girilen A değerine pozitif oranı olarak hesaplar ve ekrana yazar.

SELECT CASE DeğişkenAdı

CASE DeğişkenDeğeri1

1. Eylemsel İfade

CASE DeğişkenDeğeri2

2. Eylemsel İfade

-
-
-

CASE ELSE

Eylemsel İfade

ENDSELECT

Program akışının bir değişkenin belli değerlerine bağlı olarak gerçekleştirilecek işlemlere yönlendirilmesinde kullanılır.

Program akışı içinde,

Program:6.3.2

```
INPUT A
  SELECT CASE A
    CASE 0
      B = 0

    CASE -1
      B = 10
    CASE 1
      B = -10
    CASE ELSE
      B = 1000
  END SELECT
```

satırları, A değişken değeri 0 olarak girildiğinde B değişkeninin 0; -1 olarak girildiğinde B değişkeninin 10; 1 olarak girildiğinde B değişkeninin -10; bunların dışında bir değer girildiğinde ise B değişkeninin 1000 değerini almasını sağlar.

7. Döngüler

FOR NEXT

Döngü deyimleri ile bir program içinde belli bir satırlar kümesindeki işlemlerin istenen sayıda yinelenmesini sağlar. Daha önce bir sayaç değişkeni yardımı ve GOTO deyimini kullanılarak sağlanan yineleme işlemleri, döngü deyimleri ile daha etkin bir biçimde gerçekleştirilir. Program:6.1.2, FOR.....NEXT deyimini kullanılarak,

Program:7.1

```
INPUT A, B, N
  X = A : T = X
  FOR SAY = 2 TO N
    X = X + B : T = T + X
  NEXT SAY
  PRINT T
  END
```

biçiminde de yazılabilir.

8. Matematiksel İşlevler

Bir programlama dili ile matematiksel işlemler tanımlanırken, programcının *sin*, *cos*, *logaritma* gibi yaygın ve standart matematiksel işlevleri her seferinde programlaması düşünülemez. Her programlama dili bu matematiksel işlevleri hazır programlanmış olarak sunar. Programcı bu matematiksel işlevleri, gerektiğinde özel adları ile program sözlüğünden çağırır.

Matematiksel işlevler, program içinde

MatematikselİşlevAdı(SayısalDeğişkenAdı)

biçiminde çağrılır.

Program akışı içinde,

$$\begin{aligned} \text{PI} &= 3.141596254 : \text{A} = 4 \\ \text{X} &= \text{SQR}(\text{A}) + \text{SIN}(\text{PI} / 2) \end{aligned}$$

satırları, bellekte X adresine, 3 değerinin yazılmasını sağlar. İkinci satırdaki, **SQR** ve **SIN** ifadeleri, BASIC dilinin hazır karekök ve sin matematiksel işlevlerinin adlarıdır.

9. Dizgi İşlevleri

Alfasayısal değişken için de bir takım işlevler tanımlanabilir. Alfasayısal bir değişken değerindeki toplam simge sayısı, böyle bir değişkenin değerinden türetilebilecek başka alfasayısal değişken değerleri hazır dizgi işlevleri ile oluşturulabilir.

LEN(AlfasayısalDeğişkenAdı) deyimi, adı verilen alfasayısal değişken değerinin uzunluğunu bulur. Program akışı içinde,

$$\begin{aligned} \text{A\$} &= \text{“Atatürk”} \\ \text{B} &= \text{LEN}(\text{A\$}) \end{aligned}$$

satırları, bellekteki B adresine 7 değerinin yazılmasını sağlar.

MIDS(AlfasayısalDeğişken, Başlangıç, Sayı) deyimi, adı verilen alfasayısal değişken değerinin, içinden Başlangıç'tan itibaren Sayı kadar simgeden oluşan yeni bir alfasayısal değer oluşturur. Program akışı içinde,

```
A$ = "Atatürk"  
B = LEN(A$)  
C$ = MIDS( A$, 4, 4)
```

satırları, bellekteki C\$ adresine "türk" değerinin yazılmasını sağlar.

LEFTS(AlfasayısalDeğişken, Sayı) deyimi, adı verilen alfasayısal değişken değerinin sol tarafındaki Sayı kadar simgeden oluşan yeni bir alfasayısal değer oluşturur. Program akışı içinde,

```
A$ = "Atatürk"  
C$ = LEFTS( A$, 3)
```

satırları, bellekteki C\$ adresine "Ata" değerinin yazılmasını sağlar.

RIGHTS(AlfasayısalDeğişken, Sayı) deyimi, adı verilen alfasayısal değişken değerinin sağ tarafındaki Sayı kadar simgeden oluşan yeni bir alfasayısal değer oluşturur. Program akışı içinde,

```
A$ = "Atatürk"  
C$ = RIGHTS( A$, 4)
```

satırları, bellekteki C\$ adresine "türk" değerinin yazılmasını sağlar.

INSTR(AlfasayısalDeğişken1, AlfasayısalDeğişken2) deyimi, adı verilen 1.alfasayısal değişken değerinin içinde 2. alfasayısal değişken değerinin başladığı simgenin sıra numarasını bulur.

```
A$ = "Atatürk"  
B = INSTR(A$, "tat")
```

satırları, bellekteki B adresine 2 değerinin yazılmasını sağlar.

Alfasayısal değişkenlerin değerleri matematiksel toplama işlevi kullanılarak birleştirilebilir. Program akışı içinde,

```
A$ = "Ata" :B$ ="türk"  
C$ = A$ + B$
```

satırları, C\$ adresine "Atatürk" yazılmasını sağlar.