

Diziler

5 farklı şehirdeki otomobillerin göz önüne alalım :

	1.Şehir	2.Şehir	3.Şehir	4.Şehir	5.Şehir
Oto sayısı	1500	750	1000	2000	1800

Yukarıdaki bilgileri bilgisayar belleğinde saklamak için aşağıdaki gibi değişkenler tanımlanır ve değerler bu değişkenlere aktarılır :

```
Sehir1=1500  
Sehir2=750  
Sehir3=1000  
Sehir4=2000  
Sehir5=1800
```

Diziler

Süper ligdeki takımların durumları

Sıra	Takım Adı	O	G	B	M	A	Y	Av	P
1.	İstanbul Başakşehir	10	8	2	0	24	6	18	26
2.	Beşiktaş	10	7	3	0	21	8	13	24
3.	Galatasaray	10	6	2	2	15	8	7	20
4.	Bursaspor	10	6	2	2	13	7	6	20
5.	Fenerbahçe	10	5	3	2	21	10	11	18
6.	Konyaspor	10	3	4	3	11	11	0	13
7.	Karabükspor	10	4	1	5	13	16	-3	13
8.	Osmanlıspor FK	10	2	6	2	11	11	0	12
9.	Gençlerbirliği	10	2	5	3	9	8	1	11
10.	Gaziantepspor	10	3	2	5	12	14	-2	11
11.	Kasımpaşa	10	3	2	5	11	17	-6	11
12.	Alanyaspor	10	3	2	5	14	21	-7	11
13.	Trabzonspor	10	3	2	5	7	14	-7	11
14.	Ç.Rizespor	10	2	4	4	10	15	-5	10
15.	Kayserispor	10	2	3	5	11	17	-6	9
16.	Antalyaspor	10	2	3	5	8	15	-7	9
17.	Akhisar Bld.	10	2	3	5	6	14	-8	9
18.	Adanaspor	10	1	3	6	8	13	-5	6

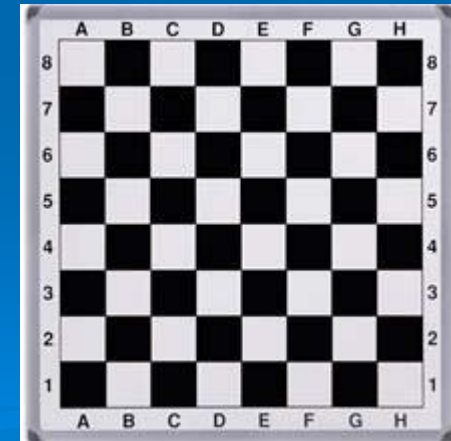
Diziler

Buradaki bilgileri bilgisayar belleğinde saklamak için ise her bilgi için değişken tanımlanması gerekmektedir:

TakimSira1=1 TakimAdi1='İstanbul Başakşehir' O1=10 G1=8 B1=2

TakimSira2=2 TakimAdi2='Beşiktaş' O2=10 G2=7 B2=3

. . .
. . .
. . .



Diziler

	oto A	oto B	oto C	oto D
1. Şehir	900	150	450	500
2.Şehir	700	600	300	550
3 Şehir	350	800	i 900	485

$n_{1,1}$ $n_{1,2}$ $n_{1,3}$ $n_{1,4}$
 $n_{2,1}$ $n_{2,2}$ $n_{2,3}$ $n_{2,4}$
 $n_{3,1}$ $n_{3,2}$ $n_{3,3}$ $n_{3,4}$

	1. kolon	2.kolon	3.kolon	4.kolon
1. satır	$N(1,1)$	$N(1,2)$	$N(1,3)$	$N(1,4)$
2. satır	$N(2,1)$	$N(2,2)$	$N(2,3)$	$N(2,4)$
3. satır	$N(3,1)$	$N(3,2)$	$N(3,3)$	$N(3,4)$

Diziler

Böylece her sayı farklı bir değişken ismiyle belirlenir. Ancak verilerin sayısı çok fazla ise, o zaman program içinde her değişkenin izlenmesi ve her değişkenle işlem yapılması problem yaratabilir.

Bu yüzden tüm sayılara ortak bir isim verip, herhangi bir tanesini dizideki numarasıyla belirlemek pratik açıdan daha uygun olacaktır. Yukarıdaki örnek için 5 farklı şehirdeki oto sayılarını temsil eden ortak isim X olsun. Matematiksel olarak,

X_1 , 1. Şehirdeki otoların sayısını

X_2 , 2.Şehirdeki otoların sayısını, vb.

gösterecektir. Böylece elemanları X_i 'lerden olan bir küme oluşturulabilir: $\{ X_1, X_2, X_3, X_4, X_5 \}$

Bu küme, şehirlerdeki otoların sayılarını gösteren bir küme olup BİR BOYUTLU DİZİ adını alır. X dizisi 5 elemanlı olup her eleman, sıralamadaki numarasıyla belirlenir. Örneğin X , dizinin ilk elemanı olup 1. Şehirdeki otoların sayısını temsil eder. FORTRAN dilinde dizinin elemanları şu şekilde yazılır : $X(1), X(2), X(3), X(4), X(5)$. Bu değişkenlerin her biri, X dizisinin bir elemanıdır (Örneğin $X(1) = 900$ 'dür).

Çoğu kez DİZİ adı verilen bir grubun elemanları olan niceliklerle çalışmak gerekebilir. Dizi, niceliklerin bir tam kümesidir.

Diziler

DIMENSION <dizi adı> (boyut) [,dizi (boyut)]....

<dizi adı> dizinin ismi programlama diline uygun olmalıdır
(boyut) dizinin boyutu/büyüklüğü

DIMENSION B(5), C(2,3), ISUM(50), X(3,2,3), Vektor(3), Matris(3,3)

- INTEGER A(20): A dizisi en fazla 20 elemanlıdır (değişkenlidir). Değişkenlerin elemanları tam sayılardan oluşur.
- REAL KAPI(2,10): KAPI dizisi de max. 20 elemanlıdır (2x10); noktalı sayı değişken elemanlarıdır.
- COMPLEX B(25): Elemanları kompleks sayılardan oluşan max. 25 elemanlı B dizisini tanımlar.
- LOGICAL X, W(10): X değişkeni mantıksal bir değişkendir. W ise en fazla 10 elemanlı bir boyutlu bir dizidir. Elemanları mantıksal değişkenlerdir.
- DOUBLE PRECISION N(3,5) : N matrisi 15 elemanlıdır (3x5). Ancak tam sayılar ikiye kazanamadığından N nin elemanları noktalı sayıları saklar.
- CHARACTER Z(20): Elemanları alfabetik karakterler olan Z dizisini tanımlar. Ancak bu şekilde her eleman sadece 1 karakter içerebilir.
- CHARACTER * 4 Z(20): Bu durumda her bir eleman 4 karakter içerebilir.

Dizi Alt ve Üst Sınırları

- Eğer bir dizinin indisi aynı zamanda DO döngüsünün de indisi olarak kullanılırsa, o zaman indisi 0 ya da negatif olan dizilerle çalışmak uygun olabilir.
- Bu durumda indis sınırları (alt sınır : üst sınır) şeklinde belirtilir.
- **Üst sınır > alt sınır** koşulu daima sağlanmalıdır. Alt sınır belirtilmemişse değeri derleyici tarafından 1 olarak kabul edilir. **Üst sınır** ise daima belirtilmelidir.
- Her boyuttaki eleman sayısı ise şu şekilde hesaplanır : **üst sınır - alt sınır + 1**. Bununla birlikte dizinin değişken indisi, **alt** ve **üst sınırlar** arasında değerler alabilir

Diziler

- DIMENSION A(10), X(0:63), ISUM(-5:4), B(0:4,10:30)
- A : En fazla 10 elemanlı, bir boyutlu reel dizidir. İndisleri 1 ile 10 arasındadır.
- X : En fazla $63-0+1 = 64$ elemanlı, bir boyutlu reel dizi. İndisi 0 ile 63 arasındadır.
- ISUM : En fazla $4-(-5)+1 = 10$ elemanlı, bir boyutlu tam sayı dizisi. İndisi -5 'den 4 'e kadar değerler alabilir.
- B : İki boyutlu noktalı sayı değişkenidir. 5 satırı ve 21 kolonu vardır. Satır indisi 0 'dan 4 'e kadar, kolon indisi ise 10 'dan 30 'a kadar değerler alabilir.

5 elemanlı bir A dizisinin sakladığı sayılar arasında en büyük ve en küçük sayıyı bulup yazan bir FORTRAN programı.

```
PROGRAM EN
IMPLICIT NONE
INTEGER :: I
REAL :: A(5), EBUYUK, EKUCUK
READ(*,*) (A(I), I=1, 5)
EBUYUK=A(1)
EKUCUK=A(1)
DO 25 I=1, 5
    IF(A(I).LE.EBUYUK) GOTO 23
    EBUYUK = A(I)
    GOTO 25
23    IF(A(I).GE.EKUCUK) GOTO 25
    EKUCUK = A(I)
25    CONTINUE
WRITE(*, 27) EBUYUK, EKUCUK
27    FORMAT(//, 'En büyük=', F7.3, 5X, 'En küçük=', F7.3,/)
STOP
END PROGRAM EN
```

Diziye Verilerin Aktarılması

C Tek boyutlu diziye kapalı döngü ile veri aktarımı

```
DIMENSION A(50)  
READ(*,*) (A(I), I=1,50)
```

C iki boyutlu diziye kapalı döngü ile veri aktarımı

```
DIMENSION A(50,50)  
READ(*,*) ((A(I), J=1,50), I=1,50)
```

C tek boyutlu diziye açık döngü ile veri aktarımı

```
DIMENSION A(10)  
DO 10 I=1, 10  
  READ(*,*) A(I)  
10 CONTINUE
```

C iki boyutlu diziye açık döngü ile veri aktarımı

```
DIMENSION A(10,20)  
DO 10 I=1, 10  
  DO 5 J=1, 20  
    READ(*,*) A(I)  
5 CONTINUE  
10 CONTINUE
```

Sıralama Yöntemleri

- Bir dizinin elemanlarını sıralama işlemi, elemanların birbiriyle karşılaştırılarak küçükten büyüğe/büyükten küçüğe doğru sıralanması demektir. İki dizi elemanının sakladıkları değerlerini birbirine aktarmak için bir ara işlem yapılması gerekir.
- Örneğin $A = 8$, $B = 5$ olsun. $A = 5$, $B = 8$ yapmak için şu adımlar izlenir:

SAKLA = A

A = B

B = SAKLA

Baloncuk Sıralama (Bubble-Sorting)

C Dizinin eleman sayısı N tane olsun

L=N-1

DO 6 K=1, L

DO 6 J=K+1, N

IF (A(K) .LE. A(J)) GOTO 6

SAKLA=A(K)

A(K)=A(J)

A(J)=SAKLA

6 CONTINUE

.....

Not : Yukarıdaki program parçası sayıları/karakterleri küçükten büyüğe doğru sıralar.

Aşağı İtme Yöntemi

C Dizinin eleman sayısı N tane olsun

M=N-1

DO 12 I=1, M

K=N-I

DO 12 J=1, K

IF (A (J) .LE. A (J+1)) GOTO 12

SAKLA = A(J+1)

A(J+1) = A(J)

A(J) = SAKLA

12

CONTINUE

.....

İndisli Değişkenlerde READ ve WRITE

C 4 elemanlı A dizisi tanımlanmış

C olsun

```
DIMENSION A(4)
```

.....

C dizi elemanlarına klavyeden

C doğrudan veri girişi yapılır

```
READ(*,*) A(1),A(2),A(3),A(4)
```

.....

İndisli Değişkenlerde READ ve WRITE

C dizi elemanlarına klavyeden

C doğrudan veri girişi döngü

C içinde yapılır

```
DIMENSION A(4)
```

```
.....
```

```
DO 10 I=1,4
```

```
10 READ( *, * ) A(I)
```

```
.....
```

İndisli Değişkenlerde READ ve WRITE

```
DIMENSION A(4)
```

```
.....
```

C dizi elemanlarına klavyeden

C doğrudan veri girişi kapalı

C döngü içinde yapılır

```
READ(*,*) (A(I), I=1,4)
```

```
.....
```


İndisli Değişkenlerde READ ve WRITE

```
DIMENSION A(4), B(5)
```

```
.....
```

```
READ( *, * ) A, B
```

İndisli Değişkenlerde READ ve WRITE

```
DIMENSION A(10,4)
```

```
.....
```

```
READ(*,*)((A(I,J),J=1,3),I=1,4)
```

C yukarıdaki kapalı döngüde

C önce A(1,1), A(1,2), A(1,3)

C sonra A(2,1), A(2,2), A(3,3)

C sonra A(3,1), A(3,2), A(3,3)

C son olarak (A(4,1), A(4,2), A(4,3))

C dizi elemanlarına değerler okunur.

C

```
READ(*,*)((A(I,J),I=1,4),J=1,3)
```

İndisli Değişkenlerde READ ve WRITE

```
DIMENSION A(10,4)
```

```
.....
```

```
WRITE(*,*)((A(I,J),J=1,3),I=1,4)
```

C yukarıdaki kapalı döngüde

C bütün elemanlar bir satırda

C ekrana serbest formatta yazılır

İndisli Değişkenlerde READ ve WRITE

C M x N ve N x P tipinde iki matrisin çarpımı.

C Sonuç matrisi C 'dir.

C234567

```
      INTEGER P
      PARAMETER ( M=4 , N= 6 , P=8)
      DIMENSION A (M,N) , B (N,P) , C (M,P)
      READ (* , *) ((A (I , J) , J=1 , N) , I=1 , M)
      READ (* , *) ((B (I , J) , J=1 , P) , I=1 , N)
      DO 100 I = 1 , M
      DO 100 J = 1 , P
      DO 100 K = 1 , N
100    C (I , J) = C (I , J) + A (I , K) * B (K , J)
      WRITE (* , *) ((C (I , J) , J=1 , P) , I=1 , M)
      END
```

İndisli Değişkenlerde READ ve WRITE

C M x N tipindeki A matrisinin
C transpozu olan T matrisi,
C N x M tipindedir.

```
PARAMETER (M=5 , N=8)
DIMENSION A (M,N) , T (N,M)
READ (* , *) A
DO 100 I=1,M
    DO 100 J=1,N
100  T (J,I) = A (I,J)
WRITE ( * , * ) T
END
```

İndisli Değişkenlerde READ ve WRITE

C N < 20 için N! 'in hesaplanması.

C234567

```
    INTEGER FAKTOR( 0 : 20 )
    READ(*,*) N
    IF(N.EQ.0) THEN
    WRITE (*,*) ' Tanım gereği 0! = 1 dir
    ENDIF
    DO 10 I=1,N 10
    FAKTOR(I-1) = I
    DO 20 L=1,N
20    FAKTOR(L) = FAKTOR(L - 1) * L
    WRITE( *, * ) N, ' ! = ', FAKTOR(N)
    END
```

Karakter dizileri

- elemanları rakamlar yerine genellikle alfabetik harfler olan dizilerdir.
- **DIMENSION** yerine **CHARACTER** tanımlama deyimini kullanılmalıdır.
- Okuma/giriş işlemleri değişkenlerdekine benzer şekilde yapılır.

Karakter dizileri

- CHARACTER A, L \Rightarrow A ve L değişkenleri birer karakter değişken ismini temsil etmektedir. Her biri tanım gereği tek karakter saklayabilir.
- CHARACTER * 4 C, D, E * 6 \Rightarrow C ve D değişkenleri 4'er, E değişkeni 6 karakterlidir.
- CHARACTER A(10) \Rightarrow A dizisi bir boyutlu karakter dizisi olup 10 tane elemanı vardır. Her eleman tek karakter taşıyabilir.
- CHARACTER * 6 D(10), E(5) * 8 \Rightarrow D dizisinin her elemanı 6 karakter, E dizisinin ise her elemanı 8 karakter taşıyabilir.
- CHARACTER * 6 A(-5 : 6) \Rightarrow A karakter dizisinin toplam $6 - (-5) + 1 = 12$ elemanı olup her eleman 6 karakter taşıyabilir.

Karakter dizileri

```
CHARACTER*6 YAZI
```

```
READ(*, 10) YAZI
```

```
10    FORMAT(A)
```

- => Bu durumda 6 karakterli YAZI değişkeninin tüm karakterleri okunur.

```
10    FORMAT(A4)
```

- => YAZI değişkeninin soldan ilk 4 karakteri okunur. Son iki karakter ise boş olarak alınır.

```
10    FORMAT(A5)
```

- => Bu durumda YAZI değişkeninin sağdan ilk 4 karakteri alınır, örneğin YAZI = abcdef ise çıktı bbcdef olur.

```
WRITE(*, 10) 'CONSTANT'
```

```
10    FORMAT(A5)
```

- => Bu durumda çıktı CONST şeklindedir.

Soru

Vektörlerin skaler çarpımını yapan bir FORTRAN programını hazırlayınız. İki vektör bileşenleri ile birlikte $\mathbf{V}_1=V_{x1}\mathbf{i}+V_{y1}\mathbf{j}+V_{z1}\mathbf{k}$ ve $\mathbf{V}_2=V_{x2}\mathbf{i}+V_{y2}\mathbf{j}+V_{z2}\mathbf{k}$ şeklinde verilebilir. Bu iki vektörün skaler çarpım sonucu $\mathbf{V}_1 \cdot \mathbf{V}_2 = V_{x1}V_{x2} + V_{y1}V_{y2} + V_{z1}V_{z2}$ şeklinde elde edilmektedir. Vektörlerin (V_1 ve V_2 yi) bileşenlerini dizilerde ($V1(1)=V_{x1}$, $V1(2)=V_{y1}$, $V1(3)=V_{z1}$ ve $V2(1)=V_{x2}$, $V2(2)=V_{y2}$, $V2(3)=V_{z2}$) tutup aynı indisli elemanları birbiri ile çarpıp bunların toplamalarını ($T=V1(1)*V2(1)+V1(2)*V2(2)+V1(3)*V2(3)$) yapan Fortran programını aşağıdaki algoritmayı kullanarak hazırlayınız ve $V_1=5\mathbf{i}-3\mathbf{j}+2\mathbf{k}$ ve $V_2=5\mathbf{i}+3\mathbf{j}+4\mathbf{k}$ değerleri için programınızı test ediniz.

1. Başla
2. $V1(3)$, $V2(3)$ dizilerini tanımlayınız
3. $V1(1)=5$
4. $V1(2)=-3$
5. $V1(3)=2$
6. $V2(1)=5$
7. $V2(2)=3$
8. $V2(3)=4$
9. $T=V1(1)*V2(1)+V1(2)*V2(2)+V1(3)*V2(3)$
10. T değişkenindeki değeri yaz
11. Son

Soru

Halen kullanılan romen rakamları arab rakamlarının aksine alfbedeki bazı karakterleri kullanır. Bu karakterler ve rakamlar

M=1000,

D=500,

C=100,

L=50,

X=10,

V=5 ve

I=1 olarak verilmektedir. LX=60, LXXV=75 örnek olarak verilebilir. Romen rakamlarında sıfır (0), negatif sayılar yoktur. Toplama, çıkarma vs işlemler yapılmaz. Bütün bunları dikkate alarak klavyeden girilen (en fazla 4 haneli) 2934, 1984, 900, 841, 485, 74 ve 60 rakamlarını romen rakamları ile ekrana yazan FORTRAN programını yazınız. Romen rakamları ve arab rakamları (0, 1, 2,...9, 10, 20, 30 ,.. 100, 200, 300, 400, ...1000, 2000, 3000, ...9000) ayrı ayrı DATA satırlarında dizilerde saklanmalı ve daha sonra kullanılmalıdır.