

CENG 111 ALGORİTMALAR VE PROGRAMLAMA

Doç. Dr. Tufan TURACI

tturaci@pau.edu.tr

- Pamukkale Üniversitesi
 - Mühendislik Fakültesi
 - Bilgisayar Mühendisliği Bölümü
- Hafta 9

İKİ BOYUTLU DİZİLER (Matrisler)

- Matris nedir?
- Matrislerin Bildirimi
- Matrislere Başlangıç Değerinin Verilmesi
- Örnekler

Çok Boyutlu Diziler

- Bir dizi aşağıdaki gibi bildirildiğinde bir boyutlu (tek indisli) dizi olarak adlandırılır. Bu tip dizilere **vektör** denir.

```
int A[5];
```

- Bir dizi birden çok boyuta sahip olabilir. Örneğin iki boyutlu y dizisi (**y matrisi**) aşağıdaki şekilde tanımlanabilir:

```
float y [3] [15];
```

- İki boyutlu diziler **matris** olarak adlandırılır.
- İlk boyuta **satır**, ikinci boyuta **sütun** denir. y matrisinin eleman sayısı $3 \times 15 = 45$ dir.

Matrislerin Bildirimi

a matrisi 3x4 tipindedir. (3 satır, 4 sütun)
a matrisinde toplam 12 eleman vardır.

	Sütun 0	Sütun 1	Sütun 2	Sütun 3
Satır 0	a [0] [0]	a [0] [1]	a [0] [2]	a [0] [3]
Satır 1	a [1] [0]	a [1] [1]	a [1] [2]	a [1] [3]
Satır 2	a [2] [0]	a [2] [1]	a [2] [2]	a [2] [3]

Sütun belirteci

Satır belirteci

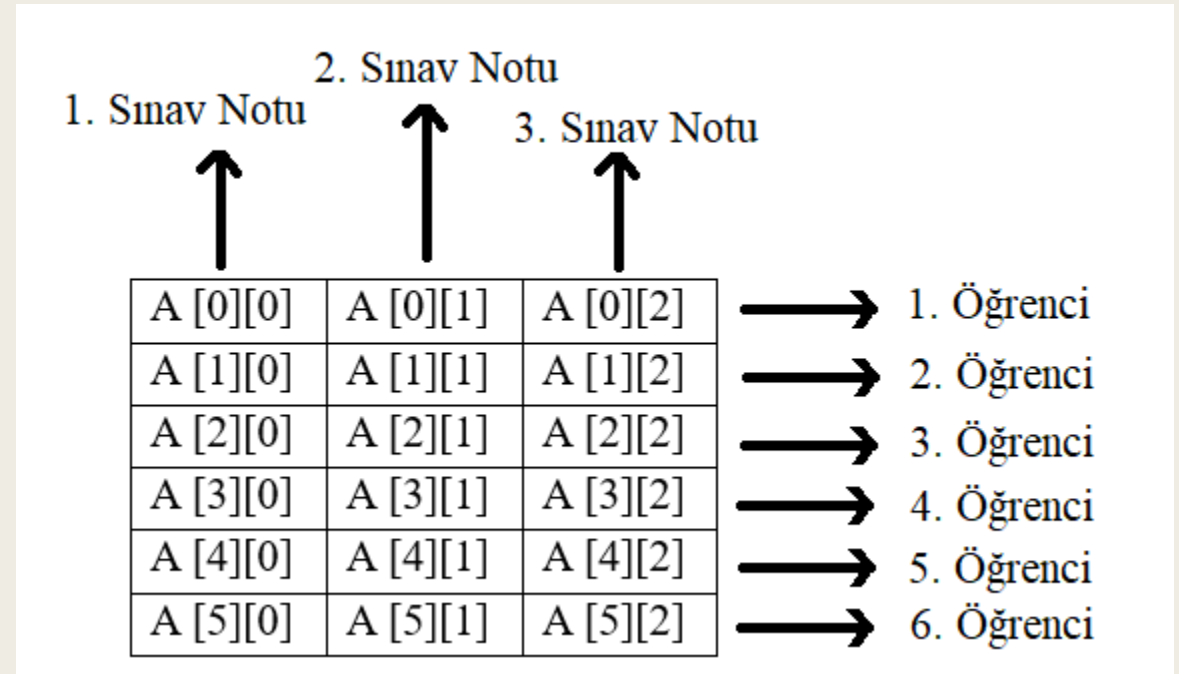
Matris ismi

- Bir matris aşağıdaki şekilde tanımlanır:

Matrisin_Veri_tipi matris_adı[satır_sayısı][sütun_sayısı] ;

- Örneğin; integer türündeki 6 adet öğrencinin 3 adet sınav notunu bellekte tutmak için aşağıdaki gibi bir matris tanımlanır

int A[6][3] ;



Matrislere Başlangıç Değeri Verme

--- Atama şeklinde:

```
int A[ 2 ][ 3 ] = { { 1, 0, 3 }, { 4, 5, 6 } },
```

```
int B[ 2 ][ 3 ] = { 1, 2, 3, 4, 5 },
```

```
int C[ 2 ][ 3 ] = { { 1, 2 }, { 4 } };
```

```
Satir satir A matrisinin elemanlari:  
1 0 3  
4 5 6  
Satir satir B dizisinin elemanlari:  
1 2 3  
4 5 0  
Satir satir C dizisinin elemanlari:  
1 2 0  
4 0 0  
-----
```

C kodu:



```
#include <stdio.h>
#include <conio.h>
int main()
{
    int A[ 2 ][ 3 ] = { { 1, 0, 3 }, { 4, 5, 6 } },
        B[ 2 ][ 3 ] = { 1, 2, 3, 4, 5 },
        C[ 2 ][ 3 ] = { { 1, 2 }, { 4 } };
    int i,j;

    printf( "Satir satir A matrisinin elemanlari:\n" );
    for ( i = 0; i <= 1; i++ ) {
        for ( j = 0; j <= 2; j++ )
            printf( "%d ", A[ i ][ j ] );
        printf( "\n" );
    }
```



```
printf( "Satir satir B dizisinin elemanlari:\n" );
    for ( i = 0; i <= 1; i++ ) {

        for ( j = 0; j <= 2; j++ )
            printf( "%d ", B[ i ][ j ] );

        printf( "\n" );
    }
    printf( "Satir satir C dizisinin elemanlari:\n" );
    for ( i = 0; i <= 1; i++ ) {

        for ( j = 0; j <= 2; j++ )
            printf( "%d ", C[ i ][ j ] );

        printf( "\n" );
    }
    getch();
    return 0;
}
```

Klavyeden sayısal değer atamak:

4 adet öğrencinin 3 adet sınav notunu bir A matrisinde saklayan ve bu matrisi ekrana yazdıran bir C programı yazınız.

```
1. ogrencinin 1. sinav notunu giriniz: 56
1. ogrencinin 2. sinav notunu giriniz: 78
1. ogrencinin 3. sinav notunu giriniz: 90
2. ogrencinin 1. sinav notunu giriniz: 87
2. ogrencinin 2. sinav notunu giriniz: 65
2. ogrencinin 3. sinav notunu giriniz: 43
3. ogrencinin 1. sinav notunu giriniz: 24
3. ogrencinin 2. sinav notunu giriniz: 58
3. ogrencinin 3. sinav notunu giriniz: 90
4. ogrencinin 1. sinav notunu giriniz: 98
4. ogrencinin 2. sinav notunu giriniz: 42
4. ogrencinin 3. sinav notunu giriniz: 78

      [1]  [2]  [3]
1. ogrenci notu[1] 56   78   90
2. ogrenci notu[2] 87   65   43
3. ogrenci notu[3] 24   58   90
4. ogrenci notu[4] 98   42   78
-----
```


C kodu:

```
#include <stdio.h>
#include <conio.h>
int main()
{
    int A[4][3],i,j;
    for (i=0;i<4;i++) {
        for (j=0;j<3;j++){
            printf ("%d. ogrencinin %d. sinav notunu giriniz: ",i+1,j+1);
            scanf("%d",&A[i][j]);
        }
    }

    printf( "           [1] [2] [3]" );

    for ( i = 0; i <4; i++ ) {
        printf( "\n %d. ogrenci notu[%d] ", i+1,i+1 );
        for ( j = 0; j <3; j++ )
            printf( "%-5d", A[i][j] );}

    getch();
    return 0;
}
```

```
scanf("%d",&A[i][j]);
yerine
scanf("%d",&x);
A[i][j]=x;
yapılabilir.
```

Örnek: Bir hastanedeki 5 hastanın günde 3 kez ateşi ölçülüyor.

a-) Her bir hastanın ortalama ateşini bulan,

b-) Her bir ölçümdeki ortalama ateş değerini bulan,

c-) Her bir hastanın günlük ölçümlerinden kaç tanesinin hastanın ortalama ateşinden fazla olduğunu bulan,
bir C programı yazınız.

```
1. hastanın 1. olcum degerini giriniz: 35.9
1. hastanın 2. olcum degerini giriniz: 36.5
1. hastanın 3. olcum degerini giriniz: 37
2. hastanın 1. olcum degerini giriniz: 39.1
2. hastanın 2. olcum degerini giriniz: 39.8
2. hastanın 3. olcum degerini giriniz: 38.2
3. hastanın 1. olcum degerini giriniz: 40.2
3. hastanın 2. olcum degerini giriniz: 39.5
3. hastanın 3. olcum degerini giriniz: 38
4. hastanın 1. olcum degerini giriniz: 35.5
4. hastanın 2. olcum degerini giriniz: 27
4. hastanın 3. olcum degerini giriniz: 37.8
5. hastanın 1. olcum degerini giriniz: 38
5. hastanın 2. olcum degerini giriniz: 38.1
5. hastanın 3. olcum degerini giriniz: 38.9
```

```
Satir satir hasta-olcum matrisinin elemanlari:
35.900002 36.500000 37.000000
39.099998 39.799999 38.200001
40.200001 39.500000 38.000000
35.500000 27.000000 37.799999
38.000000 38.099998 38.900002
1. hastanin ortalama atesi= 36.466667
2. hastanin ortalama atesi= 39.033329
3. hastanin ortalama atesi= 39.233334
4. hastanin ortalama atesi= 33.433334
5. hastanin ortalama atesi= 38.333332
1. hastanin 2. olcum degeri ortalamadan buyuktur
1. hastanin 3. olcum degeri ortalamadan buyuktur
2. hastanin 1. olcum degeri ortalamadan buyuktur
2. hastanin 2. olcum degeri ortalamadan buyuktur
3. hastanin 1. olcum degeri ortalamadan buyuktur
3. hastanin 2. olcum degeri ortalamadan buyuktur
4. hastanin 1. olcum degeri ortalamadan buyuktur
4. hastanin 3. olcum degeri ortalamadan buyuktur
5. hastanin 3. olcum degeri ortalamadan buyuktur
1. olcumdeki degerlerin ortalamasi: 37.739998
2. olcumdeki degerlerin ortalamasi: 36.180000
3. olcumdeki degerlerin ortalamasi: 37.980000
-----
```

C kodu:

```
#include <stdio.h>
#include <conio.h>
int main()
{ int i,j; float A[5][3], B[5], ort,x,top;

for (i=0;i<=4;i++)
{ for (j=0;j<=2;j++)
{
    printf("%d. hastanın %d. olcum degerini giriniz: ",i+1,j+1);
    scanf ("%f",&x);
    A[i][j]=x;
}
}
printf( "Satir satir hasta-olcum matrisinin elemanlari:\n" );
for ( i = 0; i <= 4; i++ ) //i ler satirlari, j ler sütünlari temsil eder
{
    for ( j = 0; j <= 2; j++ )
    {
        printf( "%4f ", A[ i ][ j ] );
    }
    printf( "\n" ); //satirlar deđişirken bir aşığı satıra geçer...
}
}
```





```
for (i=0;i<=4;i++)
{ top=0;
  for (j=0;j<=2;j++)
  {
    top=top+A[i][j];
  }
  ort=top/3;
  B[i]=ort; //a şıkkı
  printf ("%d. hastanın ortalama atesi= %f\n",i+1,ort);
}
```

```
// c şıkkı
for (i=0;i<=4;i++)
{ for (j=0;j<=2;j++)
  {
    if (A[i][j]>B[i]) printf ("%d. hastanın %d. olcum degeri
    ortalamadan buyuktur\n",i+1,j+1);
  }
}
```





```
// b şıkkı
for (i=0;i<=2;i++) //sütün-ateş değerleri
{ top=0;
  for (j=0;j<=4;j++) //satır-öğrenciler
  {
    top=top+A[j][i];
  }
  ort=top/5;
  printf ("%d. olcumdeki degerlerin ortalamasi:
%f\n",i+1,ort);
}

getch ();
return 0;
}
```

Soru: 3 öğrencinin bir dersten aldığı 4 farklı sınavın notları veriliyor.

En yüksek notu, en düşük notu ve her öğrencinin notlarını ortalamasını aşağıdaki şekilde yazdıran bir C programı yazınız.

```

                                [1]  [2]  [3]  [4]
ogrenciNotlari[1] 78    95    80    65
ogrenciNotlari[2] 96    90    40    56
ogrenciNotlari[3] 70    95    86    89
```

```
En dusuk not: 40
```

```
En yuksek not: 96
```

```
Ogrenci 1 icin ortalama: 79.00
```

```
Ogrenci 2 icin ortalama: 70.00
```

```
Ogrenci 3 icin ortalama: 85.00
```

```
-----
```

C kodu:

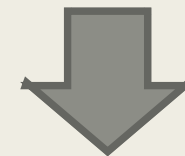
```
#include <stdio.h>
#include <conio.h>
#define OGR 3
#define SINAV 4
int main()
{
    int enk, enb,i,j,top;
    const int A[ OGR ][ SINAV ] =
        { { 78, 95, 80, 65 },
          { 96, 90, 40, 56 },
          { 70, 95, 86, 89 } };
    float ort[OGR];

    enk = 100;
    for ( i = 0; i <= OGR - 1; i++ )
        for ( j = 0; j <= SINAV - 1; j++ )
            if ( A[ i ][ j ] < enk )
                enk = A[ i ][ j ];
```



```
enb = 0;
    for ( i = 0; i <= OGR - 1; i++ )
        for ( j = 0; j <= SINAV - 1; j++ )
            if ( A[ i ][ j ] > enb )
                enb = A[ i ][ j ];

    for ( i = 0; i <= OGR - 1; i++ ){
        top = 0;
        for ( j = 0; j <= SINAV - 1; j++ ){
            top += A[i][j];
        }
        ort[i]= top / SINAV; }
```





```
printf( "          [1] [2] [3] [4]" );

for ( i = 0; i <= OGR - 1; i++ ) {
    printf( "\nogrenciNotlari[%d] ", i+1 );

    for ( j = 0; j <= SINAV - 1; j++ )
        printf( "%-5d", A[i][j] );}

printf( "\n\nEn dusuk not: %d\nEn yuksek not: %d\n", enk, enb);

for ( i = 0; i <= OGR - 1; i++ )
    printf( "Ogrenci %d icin ortalama: %.2f\n",i+1,ort[i]);

getch();
return 0;
}
```

Örnek: Klavyeden 3*3 lük iki matris giriliyor.

İki matrisin toplamını ve farkını bulan programı oluşturunuz.

```
A matrisinin 1. satir 1. sutun elemanini giriniz: 2
A matrisinin 1. satir 2. sutun elemanini giriniz: 4
A matrisinin 1. satir 3. sutun elemanini giriniz: 7
A matrisinin 2. satir 1. sutun elemanini giriniz: 8
A matrisinin 2. satir 2. sutun elemanini giriniz: 4
A matrisinin 2. satir 3. sutun elemanini giriniz: 3
A matrisinin 3. satir 1. sutun elemanini giriniz: 7
A matrisinin 3. satir 2. sutun elemanini giriniz: 9
A matrisinin 3. satir 3. sutun elemanini giriniz: 1
B matrisinin 1. satir 1. sutun elemanini giriniz: 3
B matrisinin 1. satir 2. sutun elemanini giriniz: 6
B matrisinin 1. satir 3. sutun elemanini giriniz: 8
B matrisinin 2. satir 1. sutun elemanini giriniz: 9
B matrisinin 2. satir 2. sutun elemanini giriniz: 4
B matrisinin 2. satir 3. sutun elemanini giriniz: 3
B matrisinin 3. satir 1. sutun elemanini giriniz: 2
B matrisinin 3. satir 2. sutun elemanini giriniz: 6
B matrisinin 3. satir 3. sutun elemanini giriniz: 8
```

```
Satir satir A matrisinin elemanlari:
  2   4   7
  8   4   3
  7   9   1
Satir satir B matrisinin elemanlari:
  3   6   8
  9   4   3
  2   6   8
A+B matrisinin elemanlari:
  5  10  15
 17   8   6
  9  15   9
A-B matrisinin elemanlari:
 -1  -2  -1
 -1   0   0
  5   3  -7
```

C kodu:

```
#include <stdio.h>
#include <conio.h>
int main()
{ int A[3][3],B[3][3],TOP[3][3],FARK[3][3];
  int i,j,x;
  for (i=0;i<=2;i++){
    for (j=0;j<=2;j++){
      printf ("A matrisinin %d. satir %d. sutun elemanini giriniz: ",i+1,j+1);
      scanf ("%d",&x);
      A[i][j]=x;
    }
  }

  for (i=0;i<=2;i++) {
    for (j=0;j<=2;j++) {
      printf ("B matrisinin %d. satir %d. sutun elemanini giriniz: ",i+1,j+1);
      scanf ("%d",&x);
      B[i][j]=x;
    }
  }
}
```





```
printf( "Satir satir A matrisinin elemanlari:\n" );  
    for ( i = 0; i <= 2; i++ )    //i ler satirlari, j ler sütünlari temsil eder  
    {  
        for ( j = 0; j <= 2; j++ )  
        {  
            printf( "%4d ", A[ i ][ j ] );  
        }  
        printf( "\n" ); //satirlar değişirken bir aşağı satıra geçer...  
    }
```

```
printf( "Satir satir B matrisinin elemanlari:\n" );  
    for ( i = 0; i <= 2; i++ )    //i ler satirlari, j ler sütünlari temsil eder  
    {  
        for ( j = 0; j <= 2; j++ )  
        {  
            printf( "%4d ", B[ i ][ j ] );  
        }  
        printf( "\n" ); //satirlar değişirken bir aşağı satıra geçer...  
    }
```



Toplama ve Çıkarma işlemi aşağıdaki adımda yapılıyor.



```
for (i=0;i<=2;i++)  
{  
  for (j=0;j<=2;j++)  
  {  
    TOP[i][j]=  A[i][j] + B[i][j];  
    FARK[i][j]=A[i][j] - B[i][j];  
  }  
}
```





```
printf( "A+B matrisinin elemanlari:\n" );  
    for ( i = 0; i <= 2; i++ )  //i ler satirlari, j ler sütünlari temsil eder  
    {  
        for ( j = 0; j <= 2; j++ )  
        {  
            printf( "%4d ", TOP[ i ][ j ] );  
        }  
        printf( "\n" ); //satirlar değişirken bir aşağı satıra geçer...  
    }
```

```
printf( "A-B matrisinin elemanlari:\n" );  
    for ( i = 0; i <= 2; i++ )  //i ler satirlari, j ler sütünlari temsil eder  
    { for ( j = 0; j <= 2; j++ )  
        {printf( "%4d ", FARK[ i ][ j ] );}  
        printf( "\n" ); //satirlar değişirken bir aşağı satıra geçer...  
    }
```

```
getch();  
return 0;  
}
```

Örnek: Klavyeden 3*3 lük iki matris giriliyor.

İki matrisin toplamını ve farkını bulan programı oluşturunuz.

```
A matrisinin 1. satir 1. sutun elemanini giriniz: 4
A matrisinin 1. satir 2. sutun elemanini giriniz: 1
A matrisinin 1. satir 3. sutun elemanini giriniz: 3
A matrisinin 2. satir 1. sutun elemanini giriniz: 6
A matrisinin 2. satir 2. sutun elemanini giriniz: 8
A matrisinin 2. satir 3. sutun elemanini giriniz: 7
A matrisinin 3. satir 1. sutun elemanini giriniz: 0
A matrisinin 3. satir 2. sutun elemanini giriniz: 2
A matrisinin 3. satir 3. sutun elemanini giriniz: 0
B matrisinin 1. satir 1. sutun elemanini giriniz: 1
B matrisinin 1. satir 2. sutun elemanini giriniz: 4
B matrisinin 1. satir 3. sutun elemanini giriniz: 5
B matrisinin 2. satir 1. sutun elemanini giriniz: 3
B matrisinin 2. satir 2. sutun elemanini giriniz: 7
B matrisinin 2. satir 3. sutun elemanini giriniz: 9
B matrisinin 3. satir 1. sutun elemanini giriniz: 0
B matrisinin 3. satir 2. sutun elemanini giriniz: 1
B matrisinin 3. satir 3. sutun elemanini giriniz: 2
```

```
Satir satir A matrisinin elemanlari:
  4   1   3
  6   8   7
  0   2   0
Satir satir B matrisinin elemanlari:
  1   4   5
  3   7   9
  0   1   2
C=A*B matrisinin elemanlari:
  7  26  35
 30  87 116
  6  14  18
-----
```

İki matrisin çarpımı aşağıdaki şekilde yapılır:

$$[c_{ij}] = \sum_{k=1}^{A' \text{nin Sütun Sayısı} \quad (B' \text{nin Satır Sayısı})} a_{ik} \cdot b_{kj}$$


$$4*1 + 1*3 + 3*0 = 7 \text{ (1. satır, 1. sütun elemanı)}$$

C kodu:

```
#include <stdio.h>
#include <conio.h>
int main()
{ int A[3][3],B[3][3], CARP[3][3];
  int i,j,x,top,k;

  for (i=0;i<=2;i++)
  { for (j=0;j<=2;j++)
    { printf ("A matrisinin %d. satir %d. sutun elemanini giriniz: ",i+1,j+1);
      scanf ("%d",&x);
      A[i][j]=x;
    }
  }

  for (i=0;i<=2;i++)
  { for (j=0;j<=2;j++)
    { printf ("B matrisinin %d. satir %d. sutun elemanini giriniz: ",i+1,j+1);
      scanf ("%d",&x);
      B[i][j]=x;
    }
  }
```





```
printf( "Satir satir A matrisinin elemanlari:\n" );  
    for ( i = 0; i <= 2; i++ )    //i ler satirlari, j ler sütünlari temsil eder  
    {  
        for ( j = 0; j <= 2; j++ )  
        {  
            printf( "%4d ", A[ i ][ j ] );  
        }  
        printf( "\n" ); //satirlar deđiřirken bir ařađı satıra geđer...  
    }
```

```
printf( "Satir satir B matrisinin elemanlari:\n" );  
    for ( i = 0; i <= 2; i++ )    //i ler satirlari, j ler sütünlari temsil eder  
    {  
        for ( j = 0; j <= 2; j++ )  
        {  
            printf( "%4d ", B[ i ][ j ] );  
        }  
        printf( "\n" ); //satirlar deđiřirken bir ařađı satıra geđer...  
    }
```



Çarpma işlemi aşağıdaki adımda yapılıyor.



```
for ( i = 0; i <= 2; i++ )
{
    for ( j = 0; j <= 2; j++ )
    { top=0;
      for ( k = 0; k <= 2; k++ )
      {
          x=A[i][k]*B[k][j];
          top=top+x;
      }
      CARP[i][j]=top;
    }
}
```





```
printf( "C=A*B matrisinin elemanlari:\n" );  
    for ( i = 0; i <= 2; i++ )    //i ler satirlari, j ler sutunlari temsil eder  
    {  
        for ( j = 0; j <= 2; j++ )  
        {  
            printf( "%4d ", CARP[i][j]);  
        }  
        printf( "\n" ); //satirlar degisirken bir asagi satira gezer...  
    }  
  
getch();  
return 0;  
}
```

Kaynaklar

- C: How to Program Third Edition Harvey M. Deitel ; Paul J. Deitel.
- C Programlama Dili Dr. Rıfat Çölkesen Papatya Yayıncılık.
- Problem Solving and Program Design in C, 7/E Jeri R. Hanly; Elliot B. Koffman.
- C Programlama dili; İbrahim Güney; Nobel Yayıncılık.
- Algoritma Geliştirme ve Programlamaya Giriş, Fahri Vatansever, Seçkin yayıncılık
- C Programlama Ders Notları, A. Kadir YALDIR, Pamukkale Üniversitesi ders notları.