

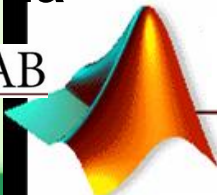
MATLAB

4.DERS

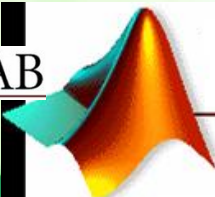
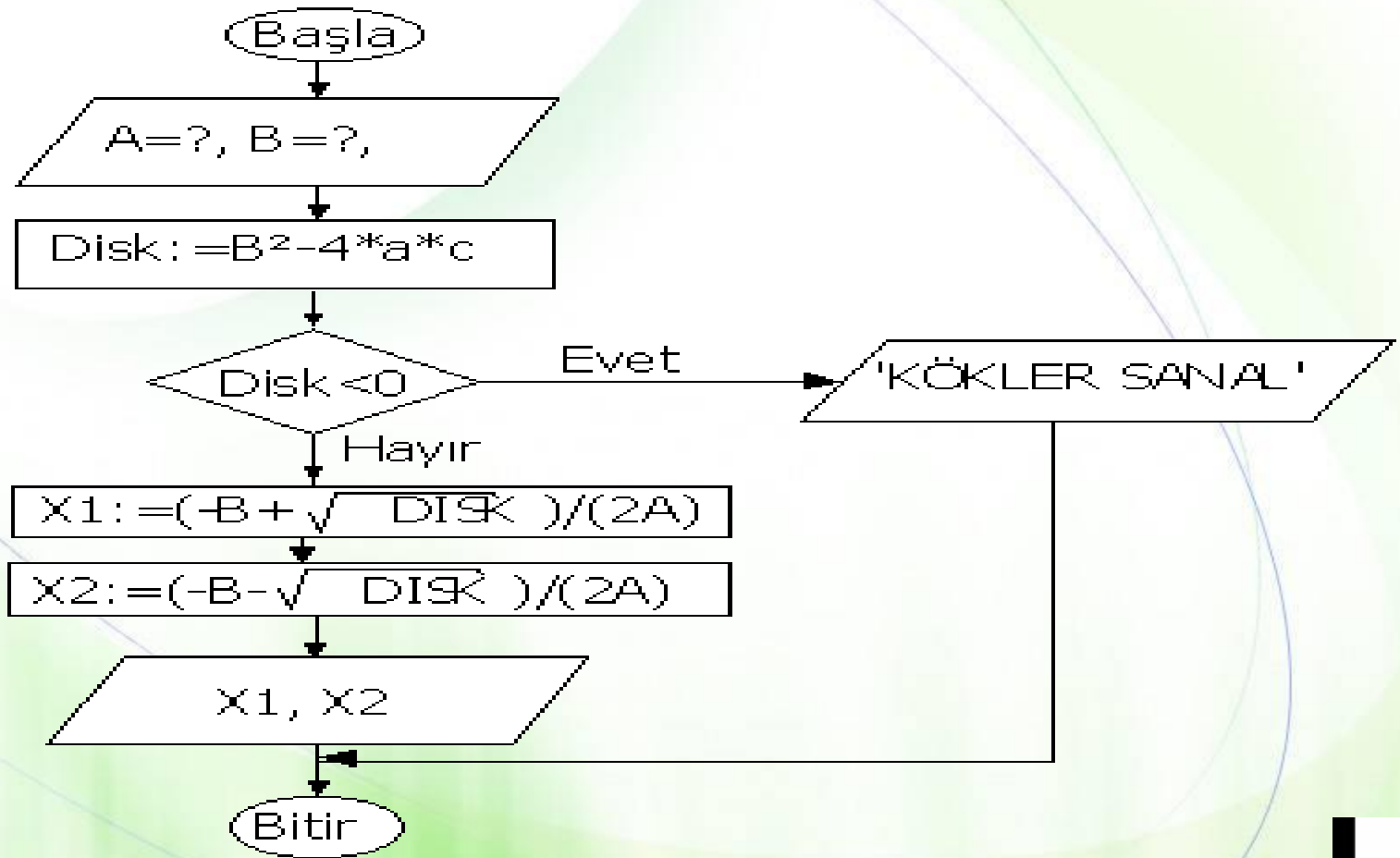


return Komutu

- Yazdığınız **MATLAB** programını herhangi bir anda (programın normalde sona erdiği noktanın haricinde - **early termination**) sona erdirmek için **return** komutunu kullanabilirsiniz.
- Bu işlem için **break** komutunu KULLANMAYINIZ.
PROGRAMINIZI ANİ SONLANDIRMAK İSTEDİĞİNİZ YER BİR DÖNGÜNÜN İÇİ İSE NE OLUR??????????
- **break** komutunu sadece **for** ve **while** döngüleri içinde kullanınız.
- Eğer **return** komutu ana program içerisinde kullanılmışsa, kontrol komut penceresindeki klavyeye geçer. Eğer **return** komutu bir fonksiyon içerisinde kullanılmışsa, kontrol bu fonksiyonu çağıran ana fonksiyona devredilir. (Fonksiyonları daha sonraki derslerimizde göreceğiz.) Bir önceki dersimizde ise **return** komutunu komut satırından çalıştırarak hata ayıklama modundan çıkmıştık.

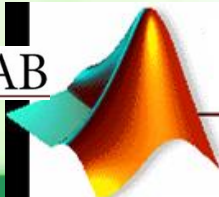


Ömek : $ax^2+bx+c=0$ şeklinde verilen 2. derece denklemin köklerini bulan programın akış diyagramını çiziniz. (Bu denklemin ikinci derece olmadığı uyarısını nasıl verirsiniz?)



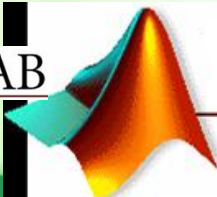
PROBLEMİN MATLAB'DA PROGRAMLANMASI

```
clc;clear;
a=input('a katsayisini giriniz= ');
b=input('b katsayisini giriniz= ');
c=input('c sabitini giriniz= ');
delta=b^2-4*a*c;
if delta<0
    disp('Kokler Sanal');
    return; %Programi Ani Sonlandir
else
    disp('Kokler Reel');
end
X1=(-b+sqrt(delta))/(2*a);
X2=(-b-sqrt(delta))/(2*a);
fprintf('1. Kok : %g \n', X1);
fprintf('2. Kok : %g \n', X2);
```



UYGULAMA : 1x5 boyutunda (1 satır ve 5 sütun) bir dizinin (**satır vektörü**) elemanlarını klavye yoluyla kullanıcıdan alan ve en sonunda bu diziyi ekrana basan **MATLAB** programı.

```
clc;clear;  
A=[ ];  
for i=1:5  
    fprintf('A(%d) = ',i);  
    A(i)=input(' ');  
end  
A
```



UYGULAMA : 2x3 boyutunda (2 satır ve 3 sütun) bir dizinin (**matris**) elemanlarını klavye yoluyla kullanıcıdan alan ve en sonunda bu diziye ekrana basan **MATLAB** programı.

```
clc;clear;
```

```
B=[ ];
```

```
for i=1:2    % Distaki for satirlar icin
```

```
    for j=1:3 % Icteki for sutunlar icin
```

```
        fprintf('B(%d,%d) = ',i,j);
```

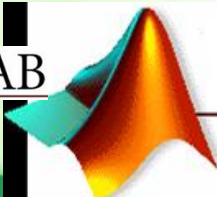
```
        B(i,j)=input(' ');
```

```
    end
```

```
end
```

```
B
```

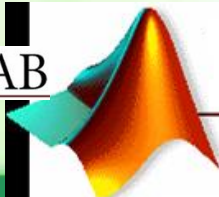
MATLAB



UYGULAMA

5 öğrencinin herhangi bir dersten aldıkları vize ve final notlarını klavye yoluyla kullanıcıdan alarak her bir öğrencinin ortalamasını hesaplayan ve aşağıdaki formata göre ekrana yazdıran bir **MATLAB** programı.

<u>Sıra</u>	<u>Vize</u>	<u>Final</u>	<u>Ortalama</u>
1. Öğrenci	35	40	37.5
2. Öğrenci	22	95	58.5
:	:	:	:



Çözüm

```
clc;clear;
```

```
vize=[ ];final=[ ];ort=[ ];
```

```
for i=1:5 %Dizilere Disaridan Veri Girisi
```

```
    fprintf('%d. Ogrencinin Vize Notu=', i); vize(i)=input(' ');
```

```
    fprintf('%d. Ogrencinin Final Notu=', i); final(i)=input(' ');
```

```
    ort(i)=(vize(i)+final(i))/2;
```

```
end
```

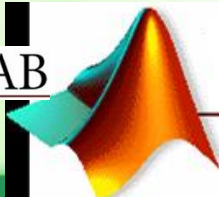
```
fprintf('Sira      Vize      Final      Ortalama\n');
```

```
fprintf('-----      -----      -----      ----- \n');
```

```
for i=1:5 %Dizi Elemanlarini Ekrana Yazdirma
```

```
    fprintf('%d. Ogrenci      %g      %g      %f\n', i, vize(i),  
    final(i),ort(i));
```

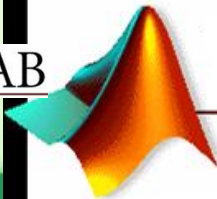
```
end
```



MATRİSLERDE İŞLEMLER

Matrislerin kendilerine ait cebirleri vardır. Ama biz özellikle matrislerin aşağıda sıralanan işlemleriyle ilgileneceğiz.

- Matrislerin Bir Skalerle Çarpımı
- Matrislerde Toplama
- Matrislerde Çıkarma
- Matrislerde Çarpma
- Matrislerin Transpozu

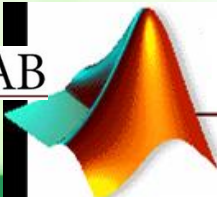


Matrislerin Bir Skalerle Çarpımı

UYGULAMA : Aşağıdaki şekliyle verilen bir **A** matrisinin her bir elemanını **2** rakamı ile çarpıp bir **C** matrisine atayan ve en sonunda bu **C** matrisini ekrana basan **MATLAB** programı.

$$A = \begin{bmatrix} 2 & 3 \\ 1 & 4 \end{bmatrix}$$

```
clc;clear;  
C=[ ];  
A=[2 3 ; 1 4];  
[m n]=size(A); %m satir ve n sutun  
for i=1:m  
    for j=1:n  
        C(i,j)=2*A(i,j);  
    end  
end  
A  
C
```



Matrislerde Toplama ve Çıkarma İşlemi

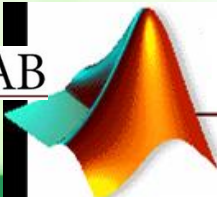
İki matrisin toplanabilmesi veya çıkarılabilmesi için boyutlarının (yani satır ve sütun sayıları) eşit olması gerekir.

$$A = \begin{bmatrix} 2 & 3 \\ 1 & 4 \end{bmatrix} \quad B = \begin{bmatrix} 3 & 6 \\ 4 & 5 \end{bmatrix}$$

İki matrisin toplamı

$$C = \begin{bmatrix} 2 & 3 \\ 1 & 4 \end{bmatrix} + \begin{bmatrix} 3 & 6 \\ 4 & 5 \end{bmatrix} = \begin{bmatrix} 5 & 9 \\ 5 & 9 \end{bmatrix}$$

```
clc;clear;  
A=[2 3; 1 4];  
B=[3 6; 4 5];  
C=[ ];  
for i=1:2  
    for j=1:2  
        C(i,j)=A(i,j)+B(i,j);  
    end  
end  
A  
B  
C
```



Matrislerde Çarpma İşlemi

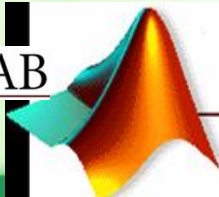
A ve **B** gibi iki matrisin çarpılabilmesi için **A** matrisinin sütun sayısının **B** matrisinin satır sayısına eşit olması gerekmektedir. **A** matrisi **mxn**, **B** matrisi **nxk** ise bu çarpma işlemi sonucunda elde edilecek **C** matrisinin boyutu **mxk** olacaktır.

Örnek

$$A = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 3 & -1 \end{bmatrix} \quad B = \begin{bmatrix} 1 & 2 \\ 1 & 5 \\ 2 & 3 \end{bmatrix}$$

$$C = A \times B = \begin{bmatrix} 5 & 15 \\ 1 & 12 \end{bmatrix}$$

```
clc;clear;
A=[1 2 1;0 3 -1];
B=[1 2; 1 5;2 3];
if size(A,2) ~= size(B,1)
    disp('Carpim Illegal');
    return;
end
C=zeros(size(A,1),size(B,2));
for i=1:2
    for j=1:2
        for k=1:3
            C(i,j)=C(i,j)+A(i,k)*B(k,j);
        end
    end
end
A
B
C
```



Matrislerde Transpoz İşlemi

Transpoz, matrislerde satır ile sütunun yer değiştirmesi işlemidir. Yani $A=2 \times 3$ 'lük bir matrisin transpozu alındığı zaman $B=A^T=3 \times 2$ 'lik bir matris elde edilir.

```
clc;clear;
```

```
A=[1 3 0; 4 2 -3];
```

```
B=[ ];
```

```
[m n]=size(A);
```

```
for i=1:n
```

```
    for j=1:m
```

```
        B(i,j)=A(j,i);
```

```
    end
```

```
end
```

```
A
```

```
B
```

A =

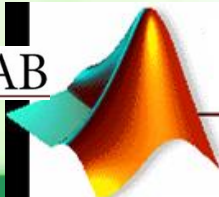
1	3	0
4	2	-3

B = A^T

1	4
3	2
0	-3

Komut satırında A' yı test ediniz.

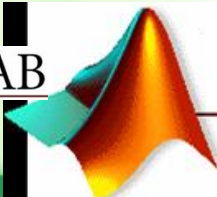
MATLAB



UYGULAMA

B=[-45 0 5 10 -91 2] dizisinin (satır vektörünün) elemanlarını tersten başka bir diziye aktaran ve bu yeni diziyi ekrana yazdıran **MATLAB** programı.

```
clc;clear;  
B=[-45 0 5 10 -91 2];  
C=[ ];  
k=length(B);  
for i=1:length(B)  
    C(k)=B(i);  
    k=k-1;  
end  
B  
C
```

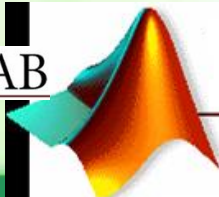


UYGULAMA

$K = \begin{bmatrix} -4 & 3 & 0 \\ 2 & 0 & 4 \end{bmatrix}$ matrisindeki sıfırların sayısını ve yerini (satır ve sütun numaralarını) ekrana basan bir **MATLAB** programı.

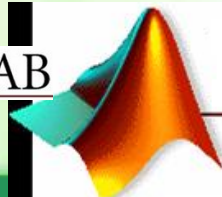
```
clc;clear;
K=[-4 3 0 ; 2 0 4 ];
[m n]=size(K);
sifirSayisi=0;
for i=1:m
    for j=1:n
        if K(i,j)==0
            sifirSayisi=sifirSayisi+1;
            fprintf('K(%d,%d) = 0 \n',i,j);
        end
    end
end
K
fprintf('K Matrisindeki SIFIR Sayisi = %d \n',sifirSayisi);
```

MATLAB



“Selection Sort” (Sıralama Algoritması)

Bu algoritma birinci elemandan başlayarak son elemana kadar, **sıralanmamış DİZİYİ** parça parça sıralar. Önce dizideki en küçük eleman bulunur ve dizinin ilk elemanı ile yer değiştirilir (**swap**). Sonraki aşamada dizinin sıralanmamış olan parçası içindeki en küçük eleman bulunur ve ikinci elemanla yer değiştirilir. Bu işlemi N defa tekrarladığımızda N elemanlı bir diziyi sıralamış oluruz.



"Selection Sort" ile Küçükten Büyüğe Sıralama

```
clc;clear;
A=round(1+49*rand(1,20)) %Sıralanmamış Diziyi Yazdır
tic %Sıralamaya Başlamadan Önce Kronometreyi Çalıştır
for i=1:(length(A)-1)
    minimum=i;
    for j=(i+1):length(A)
        if A(j)<A(minimum)
            minimum=j;
        end
    end
    %yer değiştir (swap)
    geciciDegisken=A(i);
    A(i)=A(minimum);
    A(minimum)=geciciDegisken;
end
toc %Sıralama Bittikten Sonra Kronometreyi Durdur
A %Sıralanmış Diziyi Yazdır
```

