

Matematikte Bilgisayar Uygulamaları

DERS 8 MATHEMATICA İLE PROGRAMLAMA -2 VE LİNEER CEBİR -2
MATRİSLER

DR. MURAT GEZER

Sorumuz

100 Kişiyeye (erkekler, kadınlar ve çocuklar) toplam 200 lira para verilmiştir. Her bir erkeğe 10 lira, her bir kadına 6 lira ve her bir çocuğa 1 lira verilmiş olduğuna göre erkeklerin, kadınların ve çocukların sayısını bulunuz. Mathematica ile çözümü yapınız.

Cevap:

```
In[]:= Solve[{9*x + 5*y == 100, 10*x + 6*y + z == 200}, {x, y, z}]
```

```
Out[]:= {{y -> 20-(9*x)/5, z -> 80+(4*x)/5}}
```

Ancak bu bize denklemimizde x bağlı iki durum oluşturuyor.

Mathematica'da Döngüler

İşlemlerin belli koşulda tekrar işlenmesi için kullanılmaktadır. Mathematica'da genel olarak diğer programlama dillerinde de kullanılan döngü yapıları **Do**, **For** ve **While** dir. **Sum** ve **Product** fonksiyonları diğer programlama dillerinde olmayan matematiksel özelliklerdir.

Sum ile ilgili örnek

İlk 100 tam sayının toplamı nedir?

```
In[]:=Sum[k,{k,1,100}]
```

```
Out[]= 5050
```

Do Fonksiyonu

Do[ifade,{k,kbaşlangıç,kbitiş,dk}]

Do[ifade,{n}]

Do ile ilgili örnek

Bir önceki soruyu Do[] yapısı ile yaparsak

```
In[]:=Clear[toplam,n]
      toplam=0;
      Do[toplam=toplam+n,{n,1,100}]
      Print[toplam]
```

5050

While fonksiyonu

While[koşul,ifade]

While ile ilgili örnek

Bir önceki soruyu While[] yapısı ile yaparsak

```
In[]:=Clear[toplam, n];  
    toplam = 0;  
    n = 1;  
    While[n <= 100, toplam = toplam + n; n++]  
    Print[toplam]
```


For fonksiyonu

While[koşul,ifade]

For[] ile ilgili örnek

Bir önceki soruyu For[] yapısı ile yaparsak.

```
In[]:=Clear[toplam, n];  
      For[toplam = 0; n = 1, n <= 100, toplam = toplam + n; n++];  
      Print[toplam]
```

Mathematica'da Karar yapıları

Yazdığımız Mathematica uygulaması içerisinde belli durumda işlem yapılmasını istediğimiz zaman kullandığımız yapılardır.

En kolay durumdaki kullanımı

If[durum,t,f]

Şeklindedir. Bunların dışında

Which,Switch,Piecewise yapıları da bulunmaktadır.

If[] yapısına ait örnekler

```
In[]:=x=3;  
      y=-3;  
      f[x_] := If[x > 0, "Sıfırdan Büyük", "Sıfırdan Küçük"];
```

Matris Tanımlama

`Array[fonksiyon,{n,m}]` nxm elemanlı bir matris oluşturur.

Örnekler:

```
A=Array[10 #1+#2&,{5,5}]
```

```
F[x_,y_]:=x+y^2;
```

```
Array[F,{4,6}]
```

Matris Tanımlama

Diagonalmatrix[vector] bir vektöre (listenin) elemanları köşegen olacak şekilde matris tanımlar.

Örnekler:

```
In[]:= a={2,4,19,22}
```

```
In[]:=DiagonalMatrix[a]
```

```
Out[]:= {{2,0,0,0},{0,4,0,0},{0,0,19,0},{0,0,0,22}}
```

```
In[]:= %//MatrixForm
```

```
Out[]:= 
$$\begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 19 & 0 \\ 0 & 0 & 0 & 22 \end{pmatrix}$$

```

Matris Tanımlama

IdentityMatrix[m] mxm boyutunda birim matris oluşturmaya yarar.

```
In[]:=IdentityMatrix[5];
```

```
Out[]={{1, 0, 0, 0, 0}, {0, 1, 0, 0, 0}, {0, 0, 1, 0, 0}, {0, 0, 0, 1, 0}, {0, 0, 0, 0, 1}}
```

```
In[]:=%//MatrixForm
```

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Matris İşlemleri

Matrisin mertebesini (satır ve sütun sayısı) saptamak için

Dimensions[] fonksiyonu

İki matrisi **toplamak** ve **çıkarmak** için **+** ve **-** operatörleri kullanılır

Çarpma işlemi için **.** Operatörü kullanılır

Matris İşlemleri

Bir kare matrisin determinantı

Det[]

In[]:=mat={{x,y},{z,t}};

Det[mat]

Out[]=tx-yx

Matris İşlemleri

Bir kare matrisin asal köşegenindeki elemanları:

```
In[]:=mat={{a,b},{c,d}};
```

```
Diagonal[mat]
```

```
Out[]={a,d}
```

Bir kare matrisin izi (asal köşegenindeki elemanların toplamı):

```
In[]:=mat={{a,b},{c,d}};
```

```
Tr[mat]
```

```
Out[]=a+d
```

Matris İşlemleri

Bir kare matrisin ters (determinantı sıfırdan farklı olan matrisler için geçerlidir.):

```
In[]:=Clear[a, b, c, d, mat]
      mat = {{a, b}, {c, d}};
      Inverse[mat]
```

```
Out[]= {{ $\frac{d}{-bc+ad}$ ,  $-\frac{b}{-bc+ad}$ }, {- $\frac{c}{-bc+ad}$ ,  $\frac{a}{-bc+ad}$ }}
```

Matris İşlemleri

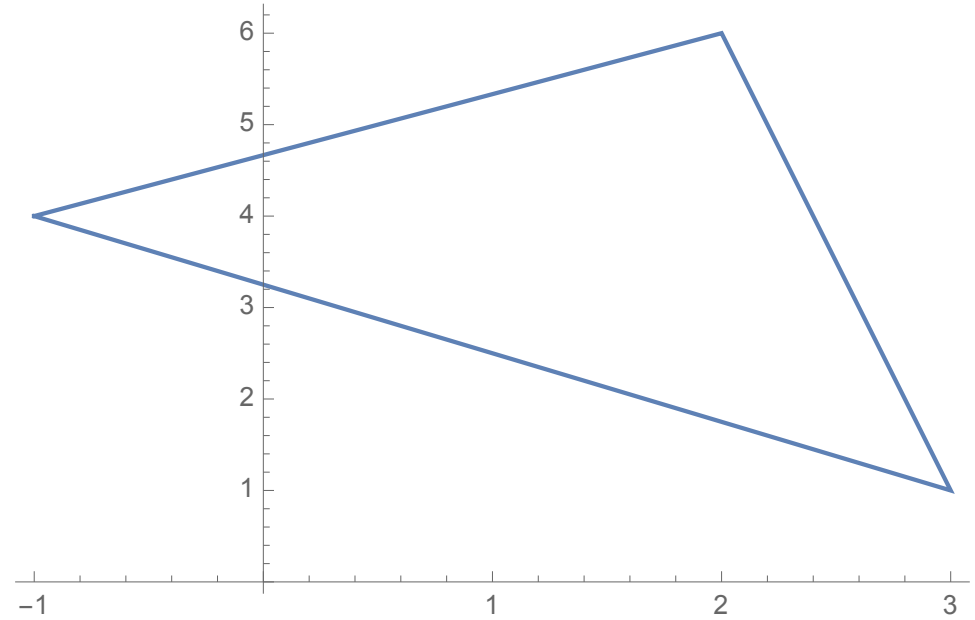
Ters kare sadece kare matrislere özgüdür. Genelleştirilmiş ters matris uygulanabilir.

Pseudoinverse[] kullanılır.

Soru

$f: R^2 \rightarrow R^2$ matris dönüşümü x-eksenine göre yansıma olsun. Köşe noktaları $(-1,4)$, $(3,1)$ ve $(2,6)$ olan T üçgeninin yansıma görüntüsünü bularak şeklini çiziniz. Her iki görüntüyü tek grafikte gösteriniz.

```
In[]:= a = {-1, 4}; b = {3, 1}; c = {2, 6};  
ListLinePlot[{a, b, c, a}]
```



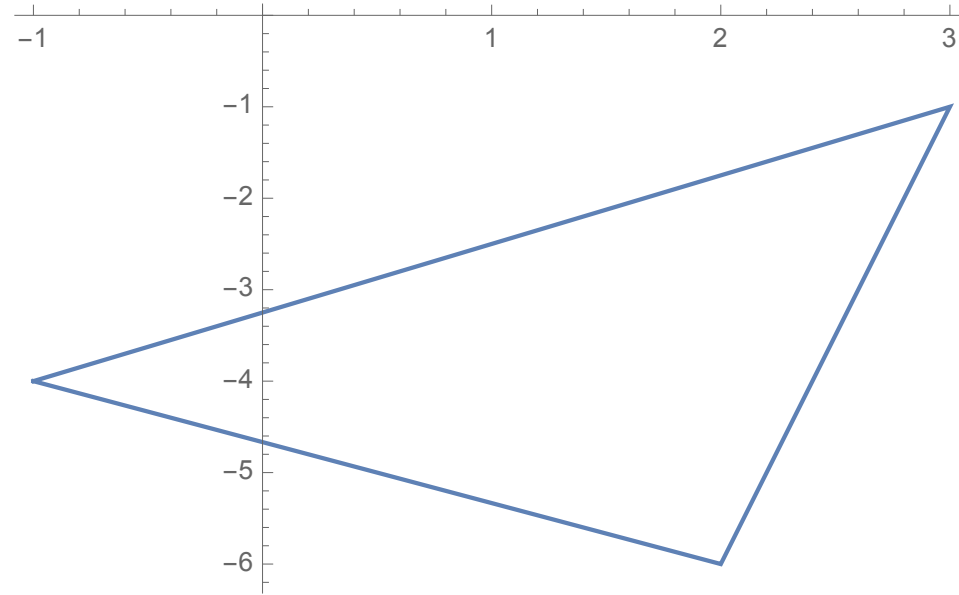
Çözüm

$f: R^2 \rightarrow R^2, f(u) = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} u$ olarak tanımlanan bir matris dönüşümü olsun. Eğer $u = \begin{bmatrix} x \\ y \end{bmatrix}$ ise bu durumda $f(u) = f\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} x \\ -y \end{bmatrix}$ olur. f matris dönüşümü R^2 de x eksenine göre yansımasıdır.

```
In[]:=Clear[a, b, c, dm, ya, yb, yc];
```

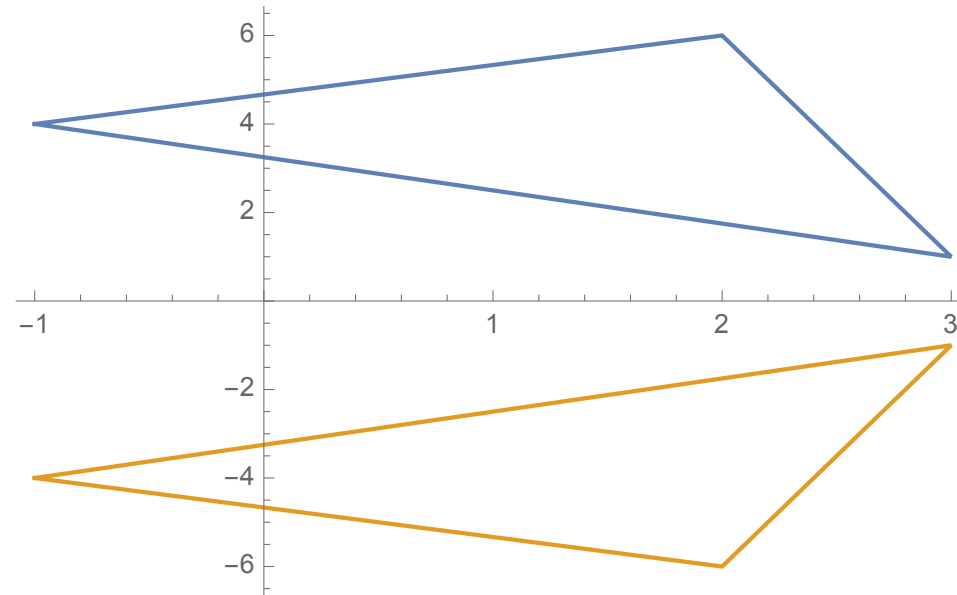
```
In[]:= a = {-1, 4};  
      b = {3, 1};  
      c = {2, 6};  
      donusummatrisi = {{1, 0}, {0, -1}};  
      ya = donusummatrisi.a;  
      yb = donusummatrisi.b;  
      yc = donusummatrisi.c;  
      ListLinePlot[{ya, yb, yc, ya}]
```

Çözüm



Çözüm

```
In[]:=ListLinePlot[{{a, b, c, a}, {ya, yb, yc, ya}}]
```



Soru

X yönünde bir sapma(shear) k bir skalar olmak üzere $f: R^2 \rightarrow R^2, f(u) = \begin{bmatrix} 1 & k \\ 0 & -1 \end{bmatrix} u$ olarak tanımlanan bir matris dönüşümü olsun. Buna göre $(0,0), (0,2), (4,2),$ ve $(4,0)$ köşelerine sahip dikdörtgeni

A-) çizdirin.

B-) $k = 2$ için oluşan yeni şekli çizdirin.

C-) Her iki şekli tek satırda ayrı ayrı olacak şekilde çizdirin.

D-) Her iki şekli tek sütunda ayrı ayrı olacak şekilde çizdirin.

Çözüm

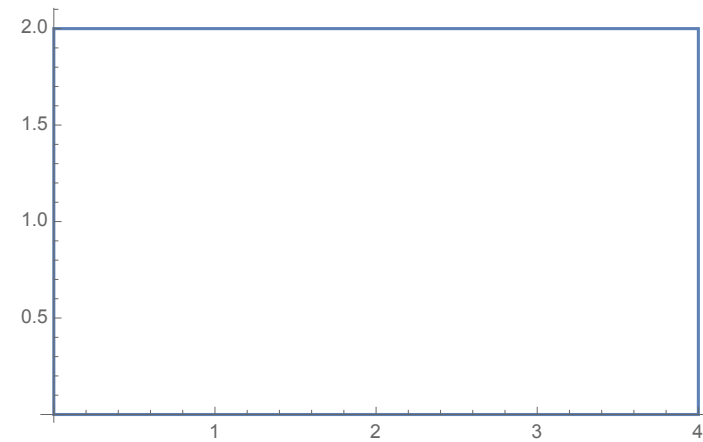
$a = \{0, 0\};$

$b = \{0, 2\};$

$c = \{4, 2\};$

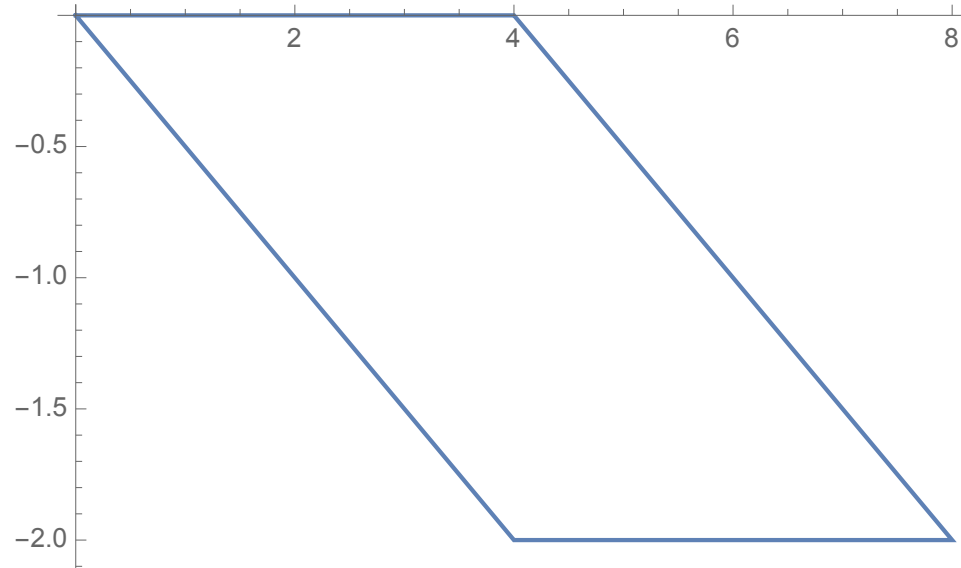
$d = \{4, 0\};$

`or=ListLinePlot[{a, b, c, d, a}]`



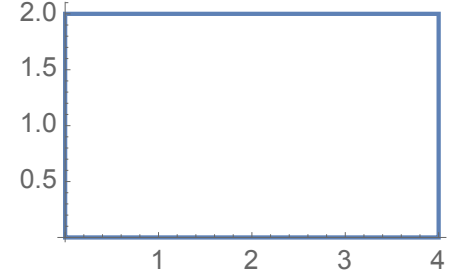
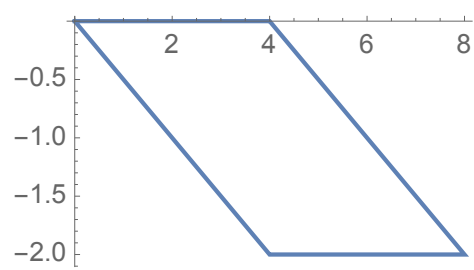
Çözüm

```
k = 2;  
dmat = {{1, k}, {0, -1}};  
ya = dmat.a;  
yb = dmat.b;  
yc = dmat.c;  
yd = dmat.d;  
dr = ListLinePlot[{ya, yb, yc, yd, ya}]
```



Çözüm

GraphicsRow[{dr, or}]



Çözüm

GraphicsColumn[{dr, or}]

