



FIRAT ÜNİVERSİTESİ
TEKNOLOJİ FAKÜLTESİ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ
BÖLÜMÜ

EET208/222 MÜHENDİSLİK YAZILIMLARI
DENEY FÖYÜ

DENEYLER

Deney 1: Matlab'a Giriş

Deney 2: Matlab Dizi ve Değişken Atamaları

Deney 3: Matlab Matematiksel Fonksiyonlarının Kullanılımının Öğrenilmesi

Deney 4: Matrisler

Deney 5: Döngü Yapıları

Deney 6: Sembolik İşlemler

Deney 7: Denklem Sistemlerinin Çözümü ve Polinomlar

Deney 8: Grafik Çizim

Deney 9: Fonksiyon Dosyası Oluşturma

Deney 10: Simulink Programı

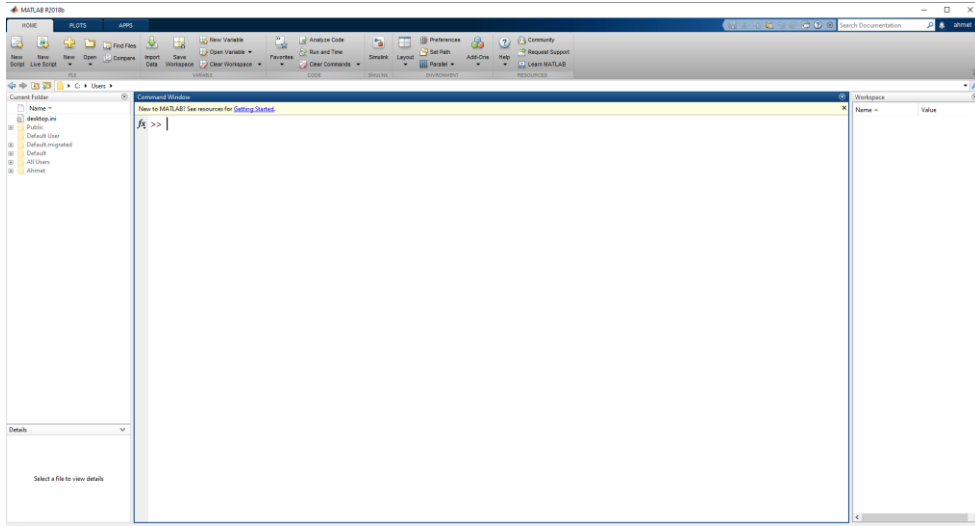
Deney No:1

Deney Adı: Matlab'a Giriş

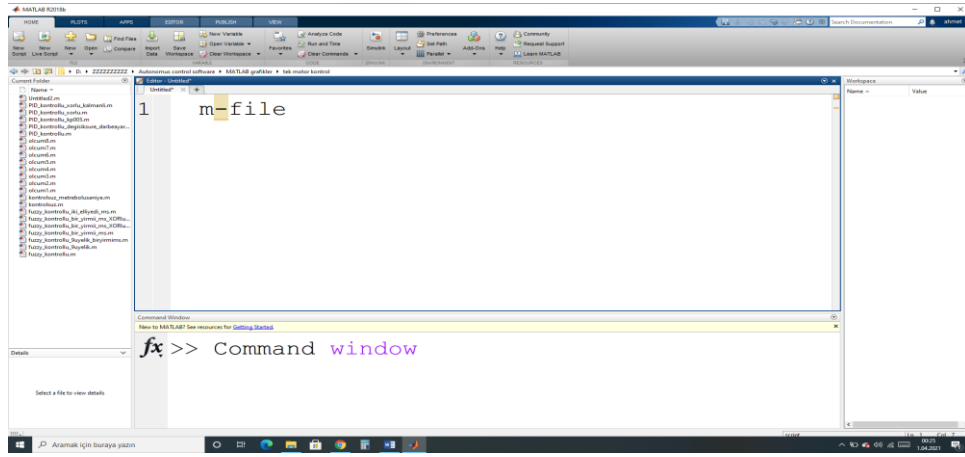
MATLAB, bilim adamları ve mühendislere, Fortran ve C gibi gelenekselleşmiş dillerde program yazmaksızın, matrislere dayalı problemleri çözmede kullanılmak üzere bir sayısal hesaplama kütüphanesi sunmak amacıyla, **MATris LABoratuvarı** (MATrix LABoratory) programı olarak tasarlanmıştır. MATLAB bir yorumlayıcıdır. Böylece sonuç, daha ziyade el tipi hesap makinelerine benzer tarzda ekranda yazılı bir metin olarak alınabilir. Neticede diğer dillerde olduğu gibi “derleme”ye ihtiyaç yoktur. Fakat programlamaya izin vermesinden dolayı da güçlü bir paket programdır. Matlab programının ilk sürümleri, Fortran diliyle hazırlanmış olmakla beraber son sürümleri C dilinde yazılmıştır.

Matlab programının genel kullanım alanları;

- Matematiksel hesaplama işlemleri
- Algoritma geliştirme ve kod yazma
- Lineer Cebir, istatistik, Fourier analizi, filtreleme, optimizasyon, integrasyon vb. konularda matematik fonksiyonlar
- 2D ve 3D grafiklerinin çizimi
- Modelleme ve simulasyon
- Veri analizi ve kontrolü



MATLAB ortamına girildiğinde görülen “>>” biçimindeki MATLAB komut girme iletisinin önüne tek tek MATLAB komut ya da fonksiyonları yazılarak çalıştırılabilir. Ancak kullanıcının, programını **File → New → m-file** menüsünden açılan text editörü içerisine yazarak ve *.m uzantısı ile kaydederek çalıştırması pratik açıdan daha uygundur.



Komut Penceresi: Matlab programı çalıştırıldığında çıkan pencereye komut penceresi, penceredeki komut yazılan kısma da komut satırı adı verilir. MATLAB yorumlayıcısı kullanıcıdan gelecek komutları kabul etmeye hazır olduğunu gösteren “>>” biçiminde bir ileti görüntüler. Örneğin, $4*25+6*52+2*99$ gibi basit matematiksel işlemi yapmak için

>> $4*25+6*52+2*99$ ifadesini yazıp **Enter** tuşuna basılınca,

ans=

610

Komut satırında yanlışların düzeltilmesi: Klavyede yer alan ok tuşları komut satırında yapılan yanlışlıkları düzeltilmesine olanak tanır. Bunlar yukarı “↑” aşağı “↓” sol “←” sağ “→”. Yukarı tuşu kullanılarak bir önceki satır tekrar görüntülenerek sağ ve sol tuşları ile yanlış yazılı yere kursör taşınarak düzenleme gerçekleştirilir.

Sonucun Ekranda Görüntülenmesini Gizleme: Bir deyim yazıp, Enter tuşuna basılınca sonuçlar ekranda otomatik olarak görüntülenir. Buna karşılık, deyim sonuna “;” ilave edilecek olursa, bu deyim ile yapılan hesaplamalar ekranda görüntülenmez.

Komutlar:

>>clc	Command window’u temizler.
>>clear	İlgili oturumda atanmış tüm değişkenleri siler.
>>clear a	Yalnızca “a” değişkenini siler.
>>demo	Matlab demosunu çalıştırır.
>>help	Yardım menüsünü açar.
>>help inv	“inv” fonksiyonunun kullanım şekli, hangi işlemi gerçekleştirdiği hakkında bilgi verir.

>>lookfor ...	İhtiyaç duyulan fonksiyonun anahtar kelimeyle aranması için kullanılır.
>>date	Gün-Ay-Yıl'ı görüntüler.
>>who	Kullanıcı tarafından tanımlanan değişkenlerin isimlerini listeler.
>>whos	Kullanıcı tarafından tanımlanan değişkenlerin isimlerini ve boyutlarını listeler.
>>what	Kullanıcı diskinin mevcut dizinindeki var olan m-dosyaların listesini verir.
<Ctrl> + <C>	Çalışan kodu durdurur.

% (Yüzde): Açıklama satırları için kullanılır.

Temel Dosya Türleri

***.m** MATLAB program dosyaları

***.fig** Grafik dosyaları ve GUI'lerin grafik parçaları

***.mat** Değişken ve matris dosyaları

İşletmenler: MATLAB deyimler içerisinde alışageldik aritmetik işletmenler ve öncelik kuralları kullanır.

İşlem	Sembol	Örnek
a+b	Toplama	2+3
a-b	Çıkarma	5-2
a*b	Çarpma	3*4
a/b	Bölme	14/7
a ^b	Üst alma	2 ³

Matematik işlemlerde öncelik hakkı: Tek bir matematiksel deyim içinde birden fazla işlem bir arada bulunabildiğine göre hangi işlemin öncelik hakkına sahip olduğunun bilinmesi yerinde olacaktır. Aşağıda MATLAB da kullanılan işlemlerde, işlemlerin öncelik listesi aşağıda verilmiştir. **1.** Parantez, **2.** Üst alma, **3.** Çarpma ve bölme, **4.** Toplama ve çıkarma

Deney No:2

Deney Adı: Matlab Dizi ve Değişken Atamaları

Diğer bilgisayar dillerinde olduğu gibi MATLAB’ın da değişken isimleri konusunda bazı kuralları vardır.

Belli Başlı Kurallar Aşağıda Olduğu Gibi Özetlenebilir.

- 1- Değişken isimleri küçük büyük harf kullanımına duyarlıdır. Buna göre aynı anlama gelen fakat farklı yazılan “orta”, “Orta”, “orTa” ve “ORTA” kelimeleri MATLAB için farklı değişkenlerdir.
- 2- Değişken isimleri en fazla 31 karakter içerebilir. Bundan fazla olanlar dikkate alınmaz
- 3- Değişken isimleri daima bir harf ile başlamalı ve bunu herhangi bir sayıda harfler, rakamlar veya alt çizgi “_” izleyebilir. **Noktalama işaretleri** değişken isminde kullanılmaz. Çünkü bunların pek çoğunun MATLAB için bir anlamı vardır.
- 4- **Rakamlar:** MATLAB rakamlar için artı veya eksi işareti ve tercihlili ondalık noktası ile birlikte alışagelmış ondalık işaretler sistemi kullanır. Bilimsel işaretler sistemi 10 tabanına göre kuvvet belirlemek için “e” harfi kullanır. Sanal rakamlar son takı olarak “i” veya “j” harfi kullanır.

3, -100, 0.0005, 9.53564, 1.456e10, 2.5e-5, 10i, -3.4j, 3e5i

Değişken atama türleri:

a.Eşitlik ifadeleri ile değişken atamaları:

degisken=değer

>>x=5

x=

5

Bir değişkene bir değer atadıktan sonra aynı değişkene farklı bir değer atamak, o değişkenin ilk değerinin silmektedir.

b. Artış Miktarı Düzenli Olan Dizilerin Değişkenlere Atanması

Düzenli artış miktarına sahip dizilerin elemanları aşağıdaki yöntemlerle atanabilir.

- Değişken = ilk_değer : artış_miktarı : son_değer

```
>>x= 1:4:20
```

```
x=
```

```
1    5    9   13   17
```

- **Linspace fonksiyonu**

Değişken = linspace(ilk_deger , son_deger , eleman_sayısı)

```
A=linspace(1,17,5)
```

```
A=
```

```
1    5    9   13   17
```

- **MATLAB Hazır Fonksiyonu**

zeros (n)	nxn boyutunda, sıfırlarda oluşan matris
zeros (n,m)	nxm boyutunda, 0'lerden oluşan matris
ones (n)	nxn boyutunda, 1'lerden oluşan matris
ones (n,m)	nxm boyutunda, 1'lerden oluşan matris
eye (n)	nxn boyutunda birim matris
eye (n, m)	nxm boyutunda birim matris
length(x)	"x" dizisinin boyutunu verir
size (x)	"x" dizisinin satır ve sütun sayısını verir

c. Bir değişkene dışarıdan bir değer atanmasını istemek

input " fonksiyonu, komut penceresinde kullanıcıdan bir değişkene bir değer girmesini isteyen bir komut görüntüler ve kullanıcının bu değeri girmesini bekler,

```
x=input('Bir değer giriniz= ');
```

Örnek:

```
>> x=input('x değerini giriniz= ')
```

x değerini giriniz= 20

x =

20

Özel Değişkenler:

ans: Komut satırındaki işlemin sonucu bir değişkene aktarılmazsa sonucun yüklendiği değişken

pi: pi sayısı

eps: epsilon

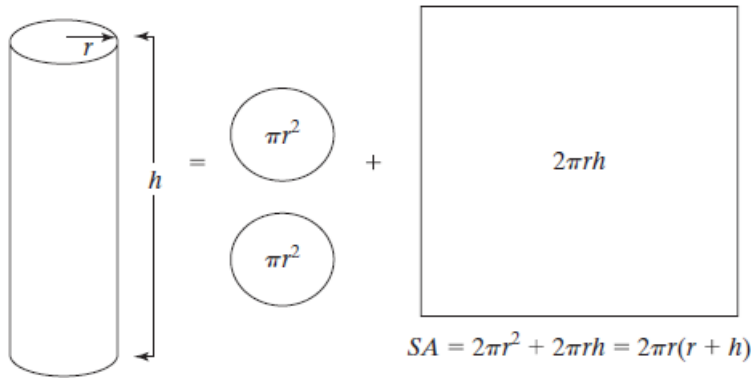
inf: sonsuz

nan(not a number): belirsizlik

UYGULAMALAR

Uygulama-2.1: Deney-2 ‘de verilen değişken atamalarının tüm türleri için ayrı ayrı Matlab üzerinde uygulamasını yapınız.

Uygulama-2.2: Aşağıda verilen silindirin yüzey alanı hesaplayan programı yazınız.



Uygulama-2.3: 1’den başlayıp 100’e kadar 30 eleman sayısına sahip olan bir dizinin elemanlarını bulan programı yazınız.

Deney No:3**Deney Adı: Matlab Matematiksel Fonksiyonlarının Kullanılımlarının Öğrenilmesi**

Matlabda yaygın olarak kullanılan matematiksel fonksiyonlarının kullanımı aşağıdaki gibidir.

Yaygın Matematiksel Fonksiyonlar			
Komut	Açıklaması	Örnek	Sonuç
abs(x)	Mutlak değer	>> abs(-3)	3
sqrt(x)	Karekök	>> sqrt(4)	2
exp(x)	Exponansiyel	>> exp(10)	22206.465
log(x)	Doğal logaritma (ln)	>> log15	2.7081
log10(x)	Logaritma	>> log10(15)	1.1761
sign(x)	Signum	>> sign(-5)	-1
rem(x,y)	x/y'nin kalanı	>> rem(15,4)	3
nthroot(x,n)	x'in n. dereceden kökü	>> nthroot(2,3)	1.2599
round(x)	Ondalıklı sayıyı en yakın sayıya tamamlar.	>> round(7.8)	8
fix(x)	Sıfıra en yakın sayıya tamamlar	>> fix(4.8)	4
floor(x)	$-\infty$ 'dan en yakın sayıya tamamlar.	>> floor(-10.2)	-11
ceil(x)	$+\infty$ 'dan en yakın sayıya tamamlar.	>> ceil(-10.2)	-10
factor(x)	x'i bölen asal sayılar	>> factor(15)	3 5
gcd(x,y)	x ve y'nin OBEB'i	>> gcd(6,21)	3
lcm(x,y)	x ve y'nin OKEK'i	>> lcm(15,6)	30
rats(x)	x'in ondalıklı yazılımı	>> rats(1.5)	3/2
factorial(x)	Faktoriyel	>> factorial(5)	120
primes(x)	x değerinden küçük asal sayılar	>> primes(12)	2 3 5 7 11

Trigonometrik Fonksiyonlar			
Komut	Açıklaması	Örnek	Sonuç
sin(x)	sinüs	>>sin(pi/3)	0.8660
cos(x)	cosinus	>>cos(pi/4)	0.7071
tan(x)	tanjant	>>tan(pi/4)	1
cot(x)	cotanjant	>>cot(pi/4)	1
asin(x)	arc sinüs	>>asin(0.5)	0.5236

NOT: Trigonometrik fonksiyonlarda açı değeri, radyan olarak işlem görür.

Kompleks İşlemler			
Komut	Açıklaması	Örnek	Sonuç
abs(z)	Kompleks sayının polar gösterimdeki genliğini bulur.	>>abs(2+3j)	3.6056
angle(z)	Kompleks sayının polar gösterimdeki açısını bulur(radyan).	>>angle(2+3j)	0.9828
conj(z)	Kompleks sayının eşleniğini bulur.	>>conj(2+3j)	2-3j
real(z)	Kompleks sayının reel kısmını bulur.	>>real(2+3j)	2
imag(z)	Kompleks sayının imajiner kısmını bulur.	>>imag(2+3j)	3

Değişkenlerin Değerlerini Görüntüleme

- **Matlab komut satırında değişkenin adının girilmesi**

```
>>x=10
```

- **disp komutu**

```
>> disp(x)
```

```
10
```

- **fprintf komutu**

```
>>fprintf( 'Açıklama %X\n',değer)
```

Burada %X kısmındaki kullanılabilecek seçenekler:

%c: Değerin tek bir karakter olduğunu gösterir.

%d: Değerin tek bir tamsayı olduğunu gösterir.

%f: Değerin tek bir ondalıklı sayı olduğunu gösterir.

%g: Değerin mümkün olan en kompakt forma sokar.

%s: Değerin bir karakter dizisi (string) olduğunu gösterir.

Ayrıca;

\n: İmleci bir alt satırın başına götürür.

UYGULAMALAR

Uygulama-3.1: Derece olarak girilen açının sinüs ve cosinüs değerini bulan bir program yazınız.

Uygulama-3.2: Aşağıdaki matematik işlemini gerçekleştiren matlab programını yazınız.

$$f = \frac{\log(ax^2 + bx + c) - \sin(ax^2 + bx + c)}{4\pi x^2 + \cos(x - 2) * (ax^2 + bx + c)} \quad x=9; a=1; b=3; c=5;$$

Uygulama-3.3: $Z = \frac{z_1*(z_2-z_3)}{z_1-z_2+3}$ denkleminde elde edilen kompleks sayının genliğini ve açısını bulan programı yazınız. ($z_1=1+3j$, $z_2=4+8j$, $z_3=2-15j$)

Deney No: 4

Deney Adı: Matrisler

Matris ve vektörler [] köşeli parantezleri ile tanımlanır.

mx1 sütun matrisi

$$B_{m,1} = \begin{bmatrix} b_{11} \\ b_{21} \\ b_{31} \\ \vdots \\ b_{m1} \end{bmatrix}$$

1xn satır matrisi

$$B_{1,n} = [b_{11} \ b_{12} \ b_{13} \ \dots \ b_{1n}]$$

mxn : m satır n sütun matrisi

$$a_{i,j} \begin{matrix} \xrightarrow{i \text{ değişir}} \\ \downarrow j \text{ değişir} \end{matrix} \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{bmatrix}_{m \times n}$$

Matris ve vektörler *.mat uzantılı olarak save komutuyla kaydedilir, load ile de istenilen yerden geri çağrılır. Örneğin, girilmiş bir a matrisini “D:\fırat” klasörüne “katsayılar.mat” olarak kaydetmek isteyelim: Bunun için aşağıdaki komut dizisi kullanılır

save D:\fırat\katsayılar a

katsayılar.mat olarak kaydedilen a matrisinin herhangi bir zamanda geri çağırılması için,

load D:\fırat\katsayılar

komut dizisi kullanılır.

Matrislerin Girilmesi

Matlab’da matrisler ve vektörler köşeli parantez ([]) ile tanımlanır. Matlab’da matrislerle ilgili bazı komutlar aşağıdaki tabloda detaylı olarak verilmiştir.

Komut	Açıklaması	Örnek	Sonuç
A=[a b c; d e f; g h k]	mxn boyutunda matrisin girilmesi	>>A=[1 3 5; 2 7 9; 4 8 6]	$A = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 7 & 9 \\ 4 & 8 & 6 \end{bmatrix}$
A=[a b c; d e f; g h k]	mxn boyutunda matrisin girilmesi	>>A=[1,3,5; 2,7,9; 4,8,6]	$A = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 7 & 9 \\ 4 & 8 & 6 \end{bmatrix}$
A=[a b c]	Satır matrisi	>>A=[1 3 5]	A=[1 3 5]
A=[a; b; c]	Sütun matrisi	>>A=[1;2;4]	$A = \begin{bmatrix} 1 \\ 2 \\ 4 \end{bmatrix}$

A(x,y)	Tanımlanan matrisin x.satır ve y.sütundaki elemanını verir.	>>A(2,3)	9
A(:,x,y))	x ve y. sütunların tamamını verir.	>>A(:,[1,3])	$A = \begin{bmatrix} 1 & 5 \\ 2 & 9 \\ 4 & 6 \end{bmatrix}$
A([x,y],:)	x ve y. satırların tamamını verir.	>>A([1,3],:)	$A = \begin{bmatrix} 1 & 3 & 5 \\ 4 & 8 & 6 \end{bmatrix}$
A(:,x)	x. sütunu verir.	>>A(:,2)	$A = \begin{bmatrix} 3 \\ 7 \\ 8 \end{bmatrix}$
A(x,:)	x. satırın tamamını verir	>>A(3,:)	$A = [4 \ 8 \ 6]$
A(:,x)=[]	A'nın x. sütununu siler.	>>A(:,1)=[]	$A = \begin{bmatrix} 3 & 5 \\ 7 & 20 \\ 8 & 6 \end{bmatrix}$
A(x,:)=[]	A'nın x. satırını siler.	>>A(3,:)=[]	$A = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 7 & 9 \end{bmatrix}$
A(x,y)=M	A'nın x.satır ve y.sütundaki değerini M değeriyle değiştirir.	>>A(2,3)=20	$A = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 7 & 20 \\ 4 & 8 & 6 \end{bmatrix}$
max(A)	A'nın her sütundaki maksimum değerini verir.	>>max(A)	4 8 9
min(A)	A'nın her sütundaki minimum değerini verir.	>>min(A)	1 3 5
[m,i]=max(A)	m=A'nın her sütunun en büyük elemanını verir. i=A'nın her sütunun en büyük elemanının satır numarasını verir.	>>[m,i]=max(A)	m=4 8 9 i=3 3 2
[m,i]=min(A)	m=A'nın her sütunun en küçük elemanını verir. i=A'nın her sütunun en küçük elemanının satır numarasını verir.	>>[m,i]=min(A)	m=1 2 5 i=1 1 1
sort(A)	Sütunları küçükten büyüğe doğru sıralar.	>> sort(A)	1 3 5 2 7 6 4 8 9
length(A)	A matrisinin satır veya sütun sayısını verir.	>>length(A)	3

det(A)	A matrisinin determinantını verir.	>> det(A)	-18
inv(A)	A matrisinin tersini verir.	>> inv(A)	1.6667 -1.2222 -1.3333 0.7778 0.06667 -0.2222
diag(A)	A matrisinin köşe elemanlarını verir.	>> diag(A)	1 7 6
rank(A)	A matrisinin rankını bulur.	>> rank(A)	3
trace(A)	A matrisinin köşe elemanlarını toplar.	>> trace(A)	14
sum(A)	Sütundaki elemanları toplar.	>> sum(A)	7 18 20
prod(A)	Sütundaki elemanları çarpar.	>> prod(A)	8 168 270
numel(A)	A matrisinin toplam eleman sayısını verir.	>> numel(A)	9
mean(A)	Herbir sütunun elemanlarının ortalamasını verir.	>>mean(A)	2.3333 6.0000 6.6667

Matrislerde Matematiksel İşlemler

Komut	Açıklaması
A+B	Toplama
A-B	Çıkarma
A*B	Çarpma

Komut	Açıklaması
A/B veya A*inv(B)	Bölme
A.*B	Her iki matrisin aynı satır ve sütunundaki elemanları çarpar.
A./B	Her iki matrisin aynı satır ve sütunundaki elemanları böler.

UYGULAMALAR

Uygulama-4.1: Kullanıcı tarafından girilen 2 matrisin çarpımını bulan programı yazınız.

Uygulama-4.2: Kullanıcı tarafından girilen bir kare matrisin tersini alan programı yazınız.

Deney No: 5

Deney Adı: Döngü Yapıları

Problem çözümlerinde komutların belirli bir koşula göre seçilmesi gerekir. Bir program içinde komutların koşula göre seçimi; belli bir koşulun doğru olması halinde belirli hesaplama takımının ve koşulun doğru olmaması halinde ise diğer bir hesaplama takımının yürütülmesine olanak sağlar. Diğer taraftan hesaplama gruplarının tekrarlı olarak yürütülmesini sağlayan döngüleri oluşturabilecek komutlara da ihtiyaç vardır. Bu nedenle MATLAB, FORTRAN, C, BASIC ve benzeri yüksek seviyeli programlama dillerinde hesaplamalar ve bu hesaplamaların hangi sırada yürütülmesi gerektiğini denetleyen mantıksal deyim fonksiyonları kullanılır. Aşağıda MATLAB'ta bulunan bazı akış denetim fonksiyonları ve bunların kullanımı hakkında bazı örnekler verilecektir.

- **if,end yapısı**

“if” yapısı bir koşulun gerçekleşmesi durumunda bir işlemi yaptırmak için kullanılır.

if koşul

İşlem

end

Örnek-1: Girilen bir sayının 10'dan büyük olması durumunda sayının karesini alan programı yazınız.

```
x=input('Bir değer giriniz= ');
```

```
if x>=10
```

```
    x=x^2
```

```
else
```

```
    x=x
```

```
end
```

- **switch,case yapısı**

Bu yapı “if” yapısına benzer. Burada daha çok sözel olarak belirtilen durumlara göre yönlendirme işlemi yapar. Bu yapısının “case” ile kullanımı aşağıdaki gibidir.

switch durum

case durum1

işlem 1

case durum2

işlem2

end

Örnek-2: Gün değişkeninin iş günü olup olmadığını karar veren programı yazınız.

```
gun=input('hangi_gun=', 's');
switch lower(gun)
    case {'pazartesi', 'sali', 'carsamba', 'persembe', 'cuma'}
        disp('is gunu')
    case {'cumartesi', 'pazar'}
        disp('TATIL')
end
```

- **for,end yapısı**

Bu döngü, bir işlemin birden fazla sayıda yaptırılmasında kullanılır.

Örnek-3: 1’den N’e kadar olan çift sayıların toplamını bulan programı yazınız.

```
N=input('bir sayi giriniz=');
toplam=0;
for i=1:1:N
    if(mod(i,2)==0)
        toplam=toplam+i;
    else
        toplam=toplam;
    end
end
toplam
```

- **while,end yapısı**

“while,end” döngüsü; belirli bir durumun gerçekleşmesi durumunda bir işlemin birden daha fazla sayıda yaptırılmasında kullanılır.

Örnek-4: 1'den N'e kadar olan sayıların toplamını bulan programı yazınız.

```
N=input('Bir değer giriniz= ');
toplam = 0;
eklenecek = 1;
while eklenecek<=N;
    toplam = toplam + eklenecek;
    eklenecek = eklenecek + 1;
end
toplam
```

UYGULAMALAR

Uygulama-5.1: Kullanıcı tarafından girilen tamsayının pozitif, negatif, tek ve çift olup olmadığını bulan programı yazınız.

Uygulama-5.2: Kullanıcı tarafından girilen bir dizinin en büyük elemanını ve bu elemanın dizinin kaçınıcı elemanı olduğunu bulan programı yazınız.

Uygulama-5.3: Girilen bir dizinin elemanları toplamını bulan programı yazınız.

Deney No: 6

Deney Adı: Sembolik İşlemler

Matlab’da integral işlemleri, sonuç belirli ise, sayısal (nümerik) olarak quad fonksiyonu ile sonuç belirsiz ise, simgesel olarak int fonksiyonu ile hesaplanabilir.

a) Belirsiz İntegral

Simgesel olarak integral hesaplamak istenildiğinde “int” komutu kullanılır.

```
>> int (f(x))      % x’e göre f(x)’in belirsiz integrali
```

Fakat bu komutu kullanabilmek için ilk olarak değişken sembolik olarak tanımlanmalıdır. Değişkenleri bildirmek için “syms” komutu kullanılır. Kullanım şekli:

```
sym x veya
```

```
x=sym('x')      dir.
```

Örnek-1: $\int x^3 + x^2 dx$ belirsiz integralin programını yazınız.

```
syms t x
f=x.^3+x.^2;
sonuc=int(f)
```

Örnek-2: $\int \frac{1}{x} dx$ belirsiz integralin programını yazınız.

```
sym x
f=1/x;
sonuc=int(f)
```

Örnek-3: $\int \sin x. dx$ belirsiz integralin programını yazınız.

```
sym x
f=sin(x);
sonuc=int(f)
```

Belirli İntegral

$\int_a^b f(x)dx$ integrali için nümerik sonuç quad ve int fonksiyonları ile bulunabilir.

quad fonksiyonunun kullanımı,

q= quad('f(x)',a,b)

biçimindedir. quad fonksiyonu hata ile f(x) fonksiyonunun a ile b arasındaki integraline yaklaşır.

Örnek-1: $\int_1^3 x^3 dx$ belirli integralinin sonucunu sayısal olarak Matlab'da bulunan quad fonksiyonunu kullanarak bulunuz.

```
>>quad('x.^3',1,3)
```

```
ans =
```

```
20
```

Örnek-2: $\int_1^{\pi} \sin(x)\cos(2x) dx$ belirli integralinin sonucunu sayısal olarak Matlab'da bulunan quad fonksiyonunu kullanarak bulunuz.

```
quad('sin(x).*cos(2*x)',0,pi)
```

```
ans = -0.6667
```

Örnek-3: $\int_0^{\pi/2} \sin 2x dx$ belirli integralin programını yazınız.

```
syms x
```

```
f=sin(x)
```

```
sonuc=int(f,x,0,pi/2)
```

Örnek-4: $\int_{-\infty}^{+\infty} e^{-x^2}$ integralin programını yazınız.

```
sym x
```

```
f=exp(-x^2);
```

```
sonuc=int(f,-inf,inf)
```

b) Türev

Türev alma işlemlerinde kullanılan fonksiyon “diff” olup simgesel yazım şekli:

```
>> diff(f(x))
```

Örnek-3: $f(x)=2x^2-5x+20$ fonksiyonunun x 'e göre türevi alınız.

```
syms x
```

```
f=(2*x^2-5*x+20);
```

```
sonuc=diff(f)
```

c) Limit

Örnek-4: $\lim_{x \rightarrow a} \frac{(x^3-a^3)}{\sin(3x-3a)}$ fonksiyonunun programını yazınız.

```
syms x a
```

```
f=(x^3-a^3)/sin(3*x-3*a);
```

```
limit(f,x,a)
```

Deney No: 7

Deney Adı: Denklem Sistemlerinin Çözümü ve Polinomlar

a) Denklem Sistemlerinin Çözümü

Matlab’da doğrusal ve doğrusal olmayan denklem sistemlerinin çözümü “solve” fonksiyonu ile gerçekleştirilir.

Örnek-1: $x^2 + 2x - 3 = 0$ denkleminin çözüm kümesini bulunuz.

```
>>syms x
```

```
>>kokler=solve(x^2+2*x-3==0,x)
```

```
kokler =
```

```
-3
```

```
1
```

Örnek-2: $x^3 + 3x^2 + 3x = 26$ denkleminin çözüm kümesini bulunuz.

```
>>syms x
```

```
>> kokler=solve(x^3+3*x^2+3*x==26,x)
```

```
kokler =
```

```
2
```

```
- (3^(1/2)*3i)/2 - 5/2
```

```
(3^(1/2)*3i)/2 - 5/2
```

Bu sonucu daha anlaşılır bir şekilde görebilmek için pretty komutu kullanılabilir.

```
>>pretty(kokler)
```

```
/      2      \
|              |
| sqrt(3) 3i   5 |
| - - - - - - - - |
|      2      2 |
|              |
| sqrt(3) 3i   5 |
| - - - - - - - - |
\      2      2 /
```

Solve komutu ile iki veya daha fazla bilinmeyenli denklemler de çözülebilir.

Örnek-3: Aşağıdaki denklem sisteminin çözüm kümesini bulunuz.

$$y+x-z=3$$

$$x-y-z=9$$

$$x+2y-3z=13$$

```
>> syms x y z
```

```
>> [x y z] = solve(y + x - z==3,x - y - z==9,x + 2*y - 3*z==13)
```

$$x = -8$$

$$y = -6$$

$$z = -11$$

b) Polinom

$P(s) = a s^4 + b s^3 + c s^2 + d s + e$ şeklindeki bir polinom Matlab programına tanıtılırken s li terimlerin katsayıları kullanılır.

$$P=[a \ b \ c \ d \ e];$$

Böylelikle katsayıları satır vektör olarak girilen polinom üzerinde Matlab'ın hazır komutlarını kullanarak, kökünün bulunması, çarpılması gibi işlemleri kolaylıkla yapılabilir.

Örnek-1: $P_1(s) = s^3 + 3s^2 + 3s + 5$ polinomunu Matlab'a tanıtınız.

```
>>P1=[1 3 3 5];
```

Örnek-2: $P_2(s) = 5s^3 + s + 1$ polinomunu Matlab'a tanıtınız.

```
>>P2=[5 0 1 1];
```

Poninomların kökleri roots komutu ile bulunur.

Örnek-3: P1 polinomunun köklerini bulunuz

```
>> P1=[1 3 3 5];
```

```
>> kokler=roots(P1)
```

kokler =

$$-2.5874 + 0.0000i$$

$$-0.2063 + 1.3747i$$

$$-0.2063 - 1.3747i$$

$P(x)$ polinomunun $s=k$ için alacağı değer “polyval” fonksiyonu ile bulunur.

Örnek-4: $P(x)=3s^3+7s^2+5s+20$ polinomunun $x=5$ için değerini hesaplayınız.

$P=[3 \ 7 \ 5 \ 20];$

polyval(P,5)

ans =

595

Polinomlarla ilgili bazı temel komutlar:

P1 ve P2 iki polinom olsun.

Komut	Açıklaması
conv(P1,P2)	İki polinomun çarpımını katsayı olarak verir.
deconv(P1,P2)	İki polinomun bölümünden bölüm ve kalanı katsayı olarak verir.
polyder(P1)	P1 polinomunun türevini alır.

Örnek-5: $P(x)=3s^3+7s^2+5s+20$ polinomunun türevini hesaplayınız.

$P=[3 \ 7 \ 5 \ 20];$

polyder(P)

ans =

9 14 5

Örnek-6: örnek-1 ve 2 de verilen P1 ve P2 polinomlarının toplamını bulunuz.

>>P1=[1 3 3 5];

>>P2=[5 0 1 1];

>> p=P1+P2

p =

6 3 4 6

UYGULAMALAR

Uygulama-7.1: s^3+3s-4 polinomunun köklerini ve türevini bulan programını yazınız.

Uygulama-7.2: Örnek 6 daki polinomların çarpımını bulunuz.

Uygulama-7.3: $P(x)=10s^4+7s^2+s+2$ polinomunun $s=8$ deki değerini bulunuz.

Deney No: 8

Deney Adı: Grafik Çizim

Matlab’da grafik çizdirmek oldukça kolaydır. 2 Boyutlu, 3 Boyutlu çizimler Matlab’da rahatlıkla yapılır. Çizim yaptırmak için vektörlere (dizilere) ihtiyaç vardır.

plot: Grafik çizimini sağlayan en temel komuttur. Komut yazılış biçimi:

plot(yatay eksen, düşey eksen, grafik sembolü, grafik rengi)

Renk	İşaretleme Biçimi	Çizgi Biçimi
y : sarı	. : nokta	- : kesintisiz (sürekli çizgi)
m : magenta	o : yuvarlak	: : nokta
c : mavi-yeşil	x : x işareti	-. : kesikli çizgi ve nokta
r : kırmızı	+ : artı işareti	-- : kesikli çizgi
g : yeşil	* : yıldız	
w : beyaz	s : kare	
k : siyah	d : elmas	
	v : aşağı üçgen	
	^ : yukarı üçgen	
	< : sola üçgen	
	> : sağa üçgen	
	p : beşgen	
	h : altıgen	

Örnek-1: $y=1+x^2$ eşitliğinin grafiğini çizdiriniz.

İlk olarak grafiğin yatay eksenini oluşturacak x değişkeni için x dizisinin oluşturulması gerekir.

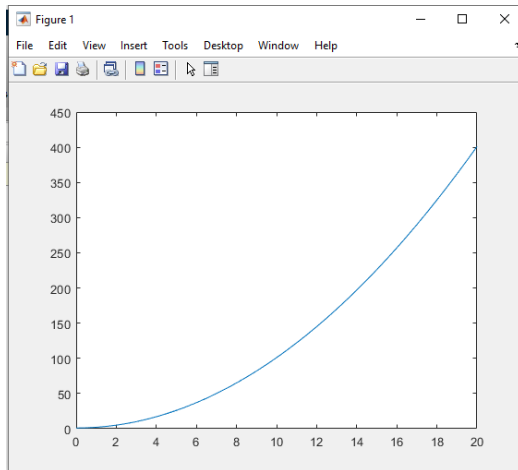
```
>>x=0:0.5:20;
```

Bu ifadeyle 0’den başlayan 0.5 adım büyüklüğü ile artan ve 20’de biten 41 elemanlı bir dizi oluşturulmuştur.

Grafiğin dikey eksenini oluşturacak y değişkeni için de y dizisini oluşturulması gerekmektedir.

```
>>y=1+x.^2;
```

```
>>plot(x,y)
```

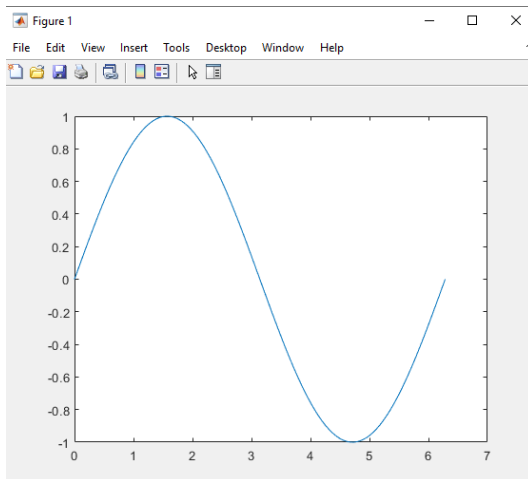


Örnek-2: Matlab da bir sinüs grafiği çizdiriniz.

```
>>wt=0:pi/360:2*pi
```

```
>>y=sin(wt)
```

```
>>plot(wt,y)
```



hold on: Ekranda aynı anda birden fazla grafik görmeyi sağlar.Yukarıdaki örneklerde iki değişken de arka arkaya yazıldığında son yazılanın grafiği ekrana gelir ancak ilkinin yazıldıktan sonra bu komut kullanılırsa ekranda kalmaya devam eder.

Örnek-3: örnek-1 ve örnek-2 deki grafikleri birlikte çizdiriniz.

```
x=0:0.5:7;
```

```
y=1+x.^2;
```

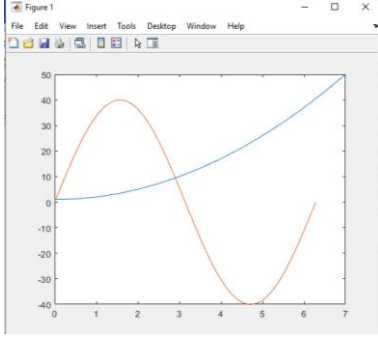
```
plot(x,y)
```

```
hold on
```

```
wt=0:pi/360:2*pi
```

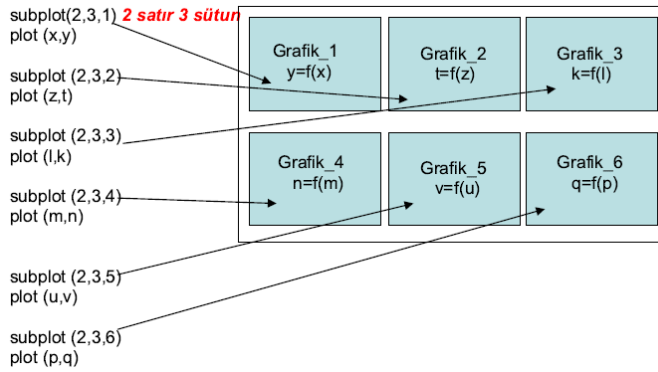
```
y=40*sin(wt)
```

```
plot(wt,y)
```



grid on: Grafik düzlemindeki bölmelendirme çizgilerinin görüntülenmesi için “grid on” ve bu özelliğin kapatılması için “grid off” fonksiyonları kullanılır.

subplot: Aynı düzlem üzerinde aynı eksen takımlarını kullanarak tek bir grafik penceresinde birden fazla grafik çizdirmek kullanılır.



Örnek-4: Yukarıdaki örneklerde verilen 2 grafiği tek bir pencerede ayrı ayrı çizdiriniz. Sinüs eğrisinde bölmelendirme çizgisi ekleyiniz

```
x=0:0.5:7;
y=1+x.^2;
subplot(2,1,1);
plot(x,y)
wt=0:pi/360:2*pi
y=40*sin(wt)
subplot(2,1,2);
plot(wt,y)
grid on
```

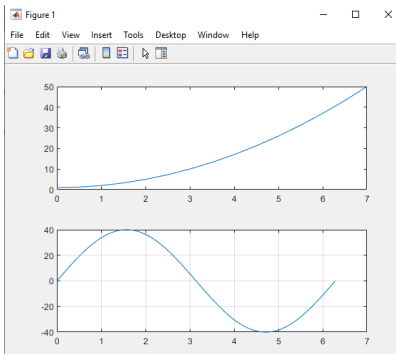


figure: Grafikleri farklı grafik pencerelerinde oluşturmak için “figure” fonksiyonu kullanılır.

Kullanım şekli;

figure(n): n pencerenin numarasıdır

Örnek-5: $P(t)=3t^3+7t^2+5t+20$ polinomunun, zaman eksenine göre çizdiriniz. (t:1:10:100)

t=1:10:100;

P=[3 7 5 20];

Y=polyval(P,t)

plot(t,Y, 'r-')

grid on

Örnek-6: sinüs ve cosinüs fonksiyonlarının 0-2*pi aralığında değişimlerini farklı grafik pencerelerinde çizdiriniz.

t=0:pi/100:2*pi;

y1=sin(t);

y2=cos(t);

figure(1)

plot(t,y1,'r-')

figure(2)

plot(t,y2,'b--')

title: Grafiğe başlık eklemek için kullanılır.Kullanım şekli:

title('başlık');

xlabel: x ekseninin ismi yazılır. Kullanım şekli:

xlabel('x ekseninin ismi')

ylabel: y ekseninin ismi yazılır. Kullanım şekli:

ylabel('y ekseninin ismi')

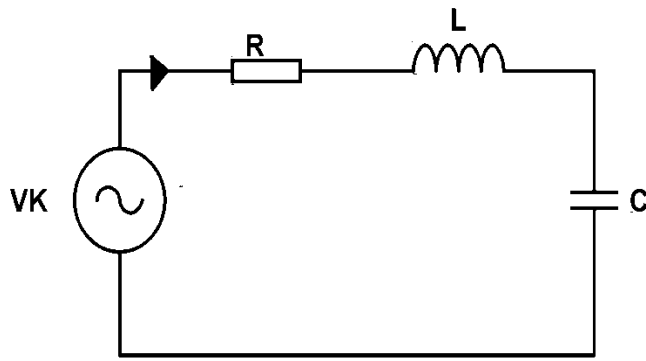
UYGULAMALAR

Uygulama-8.1: Ekranı 2 parçaya bölüp $\pi/2$ - $\pi/2$ aralığında sin ve cos fonksiyonlarının değişimini çizdiriniz.

Uygulama-8.2: Aynı ekran üzerinde 0- π aralığında sinus ve cosinus fonksiyonlarının değişimini çizdiren programı yazınız.

Uygulama-8.3: Seri RC devresinde V_c geriliminin 0-10ms arasında değişimini çizdiriniz.
($R=30$; $C=50\mu F$; $V_k=5V$; $V_c= V_k (1-e^{-t/\tau})$; $\tau=RC$)

Uygulama-8.4:



Yandaki devrede $R=20\Omega$, $C=0.01F$, $L=0.5H$ ve $V_k=20*\sin(10t)$ olarak verilmiştir. Devredeki akımı ve elemanların gerilimlerini kompleks olarak bulacak ve çizim ekranını dörde bölerek sırasıyla devreden geçen akımın, V_R , V_L , V_C ' nin zamana göre değişimini 0-1 sn aralığında çizdirecek bir m-file oluşturunuz.

Deney No:9

Deney Adı: Fonksiyon Dosyası Oluşturma

Matlab fonksiyonları, kullanıcının bir başka programa gerek duymaksızın temel işlemleri kolayca yapabilmesini sağlar. Fonksiyonlar m-dosyası içinde saklanır ve bu dosyalara, fonksiyon dosyaları adı verilir. Kullanıcılar kendi fonksiyonlarını geliştirebilirler. Fonksiyonlar da bilgisayar programlarıdır. Bilgisayarlar yardımıyla çözülmeye çalışılan problemler fonksiyonlar sayesinde yönetilmesi daha kolay, küçük parçacıklara bölünürler. Bu metoda “Böl ve Yönet” ya da “Divide and Conquer” adı verilir. Her fonksiyon:

- Kendine özgü bir isme sahiptir.
- Genellikle kendine, üzerinde işlem yapacağı bir argüman (parametre) ya da argümanlar (parametreler) alır.
- Genellikle geriye bir değer döndürür. (skaler, vektör ya da matris.)

Fonksiyon m-dosyaları:

Bir fonksiyon m-dosyasının iki bileşeni vardır:

1 – Fonksiyonun imzası veya prototipi (İlk satırda tanımlanır.)

2 – Fonksiyonun tanımı (yapması gereken iş) (İkinci satırdan başlar ve devam eder.)

Fonksiyon imzası veya prototipi

`function cikisParametresi = FonksiyonAdi (girisParametreleri 1, 2, ...n)`

`function [cikisParametreleri 1, 2,..., n] = FonksiyonAdi (girisParametreleri 1, 2, ...n)`

`function cikisParametresi = FonksiyonAdi ( )`

`function cikisParametresi = FonksiyonAdi`

`function FonksiyonAdi (girisParametreleri 1, 2, ...n)`

Fonksiyon m-dosyalarının ilk satırı MUHAKKAK yukarıda tanımladığımız gibi fonksiyonların imzası veya prototipi dediğimiz satırlardan oluşmalıdır. Bir fonksiyonu yazmaya başlamadan önce onun imzası hakkında iyice düşünmeliyiz. (Giriş parametresi(leri) almalı mı, geriye bir değer döndürmeli mi?)

Fonksiyon M-Dosyaları Oluşturulurken Dikkat Edilmesi Gereken Hususlar:

- 1 - Her MATLAB fonksiyonu function anahtar kelimesi ile başlamalıdır.
- 2 – Fonksiyon Adi m-dosyasına verilen isimle aynı olmalıdır.
- 3 - Bir MATLAB fonksiyonu komut penceresinden fonksiyon adı ve varsa eğer parantez içerisinde birbirlerinden virgüllerle ayrılmış parametrelerle çağrılmalıdır.
- 4 - Parametre aktarımı olması durumunda alt ve ana programda eşit sayıda giriş parametresi olmalıdır.

Örnek-1: Kendisine gönderilen iki sayının toplamını ve çarpımını bularak geri gönderen bir fonksiyon oluşturunuz.

```
function [x,y]=topcarp(a,b)
x=a+b;
y=a*b;
end
```

Örnek-2:

```
function [y1,y2,y3]=fonksiyon(a,b,c);
y1=a*sin(b);
y2=max(b);
y3=upper(c);
%Bu fonksiyon adı ile kaydedilip aşağıdaki gibi çağırılıp çalıştırılabilir.
%[x,y,z]=fonksiyon(3,[2 4],'firat')
```

UYGULAMALAR

Uygulama-9.1: Kendisine gönderilen iki polinomun çarpımını ve türevlerinin çarpımını çıkış olarak geri gönderen bir m-function yazınız.

Uygulama-9.2: Kendisine gönderilen 2 dizinin toplam uzunluğunu bularak geri gönderen bir m-function oluşturunuz.

Uygulama-9.3: Kendisine gönderilen 2 dizinin boyutları eşit ise toplayıp geri gönderen değilse ekrana bir uyarı yazarak kısa diziyi geri gönderen bir m-function oluşturunuz.

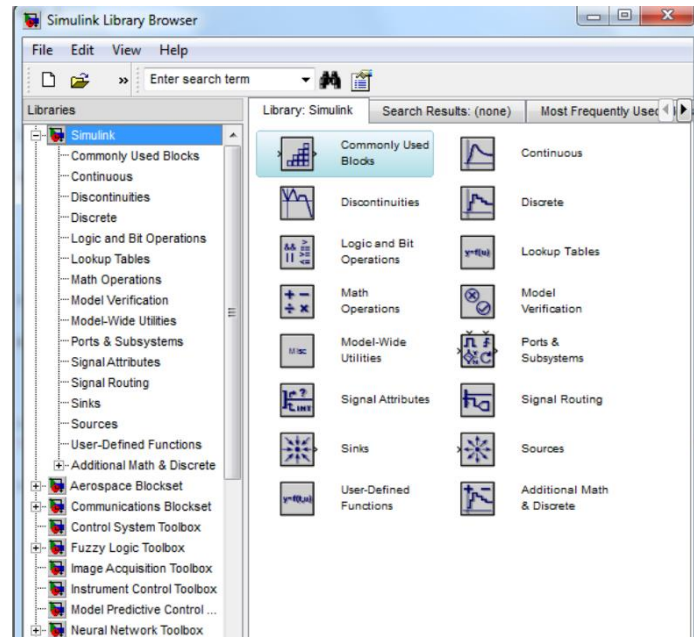
Deney No:10

Deney Adı:Simulink Programı

SIMULINK; MATLAB programı (*.m dosyası) yazarak bir dinamik sistemin cevabının bulunması ve blok şemalarının fonksiyonlarla indirgenerek kontrol sisteminin incelenmesi yerine doğrudan blok şemalar ile kontrol sistemi çizilerek analizini sağlayan bir programdır. MATLAB/SIMULINK ortamına geçmek için komut satırından,

```
>> simulink
```

yazılmalı ya da Simulink düğmesi tıklanmalıdır. Simulink ortamında temel simulink blok kütüphaneleri ile birlikte çok sayıda toolbox blok kütüphaneleri mevcuttur. Temel simulink blok kütüphanelerinden ilk aşamada Continuous, Sinks, Sources, Math Operations kütüphaneleri önemlidir.

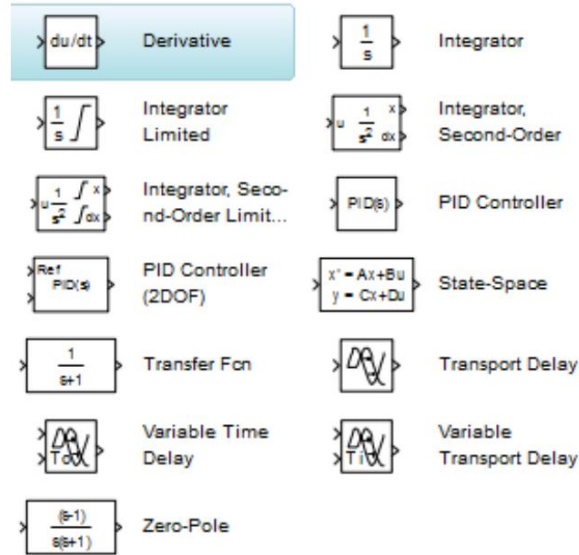


Yeni bir Simulink dosyası oluşturmak için SIMULINK den File-New seçilerek boş bir simulink ortamı açılmalıdır. Bu boş ortamda Simulink ile bir kontrol sisteminin blok şemasını oluşturmak için yukarıdaki şekilde görülen kütüphane blokları seçildikten sonra içeriğindeki ilgili bloklar sürüklenerek taşınmalı ve bağlantıları yapılmalıdır.

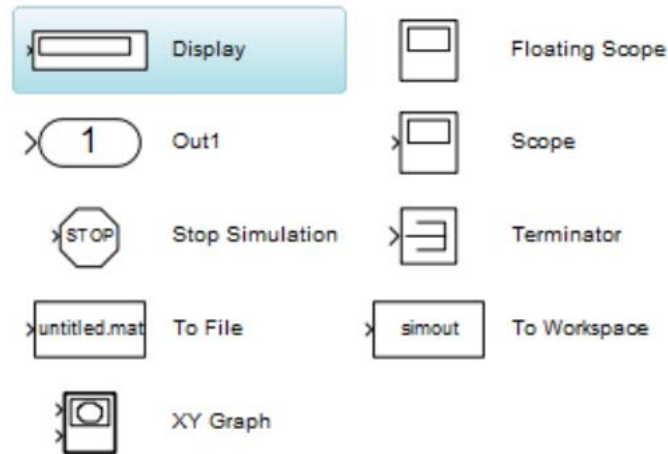
Model kurulduktan sonra dosyaya isim verilerek kaydedilebilir. Kurulan model, Simulation-start ile çalıştırılır. Bazı önemli blok kütüphaneleri ve bloklar aşağıda verilmiştir.

Math Operations Kütüphanesi çarpma, bölme, toplama, işaret alma, karekök alma vs. gibi çeşitli aritmetiksel operatör bloklarını ihtiva eder.

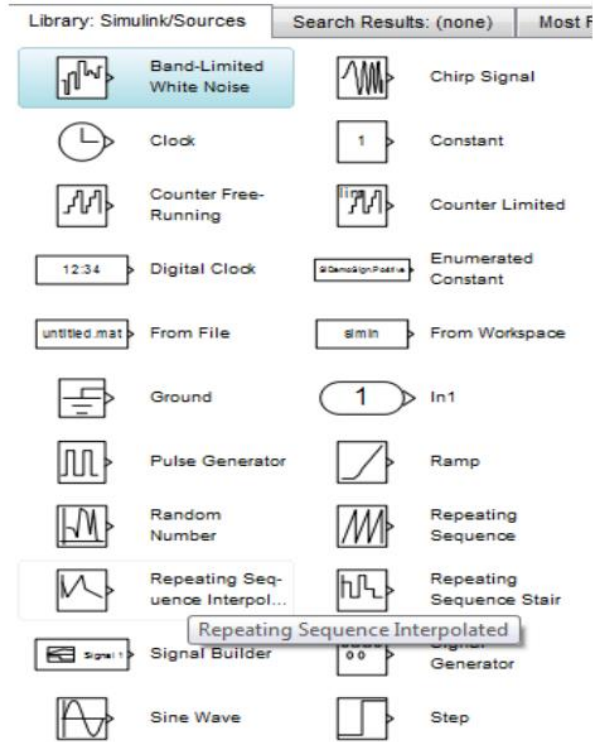
Continuous kütüphanesi türev, integral, sistemlerin transfer fonksiyonu ve durum denklemi modellerini ihtiva eder.



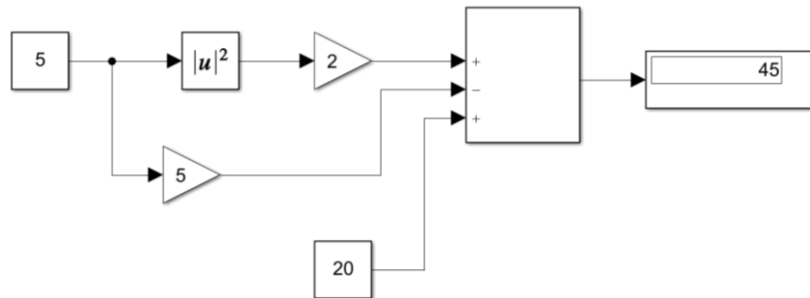
Sinks kütüphanesi çeşitli osilaskoplar, blok çıkışı, dosya ve MATLAB komut satırına yazma bloklarını içerir



Sources sabit, sinüs, pulse vs. gibi çeşitli sinyalleri içerir.



Örnek-1: $x=5$ için $f(x)=2x^2-5x+20$ denkleminin değerini bulan modeli oluşturunuz.



Örnek-2: Tepeden tepeye 10 volt genliğe, 1 Hz frekansa sahip bir sinüs sinyalinin grafiğini çizdiriniz.

