

**FIRAT ÜNİVERSİTESİ
TEKNOLOJİ FAKÜLTESİ
ELEKTRİK VE ELEKTRONİK MÜHENDİSLİĞİ
BÖLÜMÜ
2018-2019 BAHAR DÖNEMİ
EET208/222 MÜHENDİSLİK YAZILIMLARI
DENEY FÖYÜ**

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
- Algortima 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

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

Komutlar:

>>clc	Command window’utemizler.
>>clear	İlgili oturumda atanmış tüm değişkenleri isler.
>>clear a	Yalnızca “a” değişkenini iler.
>>intro	Programın genel özelliklerinin incelenmesini sağlar.
>>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.
>>save d a	“a” değişkenini d dosya ismiyle mat uzantılı olarak kaydeder.
>>load d	“a” değişkenini d dosyasından geri çağırır.
>>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.

% (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.
- 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 tercihli 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 sütun sayısını 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

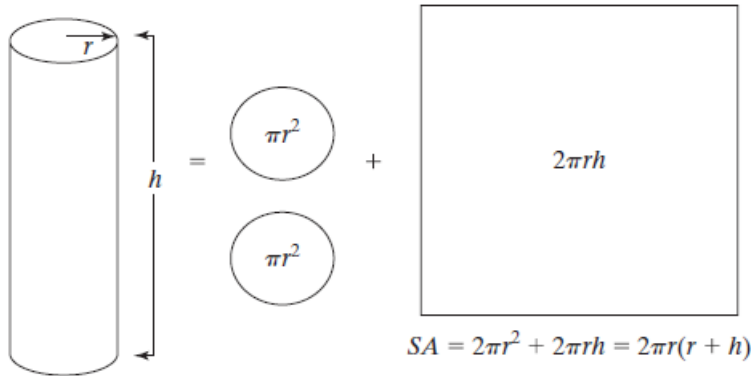
inf: sonsuz

nan(not a number): belirsizlik

UYGULAMALAR

Uygulama-1: Deney-2 'de verilen değişken atamalarının tüm türleri için ayrı ayrı Matlab üzerinde uygulamasını yapınız.

Uygulama-2: Aşağıda verilen silindirin yüzey alanı hesaplayan programı yazınız.



Uygulama-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)	Ondalıkli 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 aç ı değeri, radyan olarak iş lem 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-1: Derece olarak girilen açının sinüs ve cosinüs değerini bulan bir program yazınız.

Uygulama-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: $Z = \frac{z_1*(z_2-z_3)}{z_1-z_2+3}$ 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 ve Döngü Yapıları

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.	$\gg A(2,3)$	9
$A(:,[x,y])$	x ve y. sütunların tamamını verir.	$\gg A(:,[1,3])$	$A = \begin{bmatrix} 1 & 5 \\ 2 & 9 \\ 4 & 6 \end{bmatrix}$
$A([x,y],:)$	x ve y. satırların tamamını verir.	$\gg A([1,3],:)$	$A = \begin{bmatrix} 1 & 3 & 5 \\ 4 & 8 & 6 \end{bmatrix}$
$A(:,x)$	x. sütunu verir.	$\gg A(:,2)$	$A = \begin{bmatrix} 3 \\ 7 \\ 8 \end{bmatrix}$
$A(x,:)$	x. satırın tamamını verir	$\gg A(3,:)$	$A = [4 \ 8 \ 6]$
$A(:,x) = []$	A'nın x. sütununu siler.	$\gg A(:,1) = []$	$A = \begin{bmatrix} 3 & 5 \\ 7 & 20 \\ 8 & 6 \end{bmatrix}$
$A(x,:) = []$	A'nın x. satırını siler.	$\gg A(3,:) = []$	$A = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 7 & 9 \end{bmatrix}$
$A(x,y) = M$	A'nın x.satır ve y.sütundaki değerini M değeriyle değiştirir.	$\gg A(2,3) = 20$	$A = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 7 & 20 \\ 4 & 8 & 6 \end{bmatrix}$
$\max(A)$	A'nın her sütundaki maksimum değerini verir.	$\gg \max(A)$	4 8 9
$\min(A)$	A'nın her sütundaki minimum değerini verir.	$\gg \min(A)$	1 3 5
$[m,i] = \max(A)$	m=A'nın her sütunun en büyük elemanını verir. i=A'nın her sütunun en büyük elemanının	$\gg [m,i] = \max(A)$	m=4 8 9 i=3 3 2

	satır numarasını verir.		
[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 0.4444 -1.3333 0.7778 -0.0556 0.06667 -0.2222 -0.0556
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.

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-1: Kullanıcı tarafından girilen 2 matrisin çarpımını bulan programı yazınız.

Uygulama-2: Kullanıcı tarafından girilen bir kare matrisin tersini alan programı yazınız.

Uygulama-3: Kullanıcı tarafından girilen tamsayının pozitif, negatif, tek ve çift olup olmadığını bulan programı yazınız.

Uygulama-4: Kullanıcı tarafından girilen bir dizinin en büyük elemanını ve bu elemanın dizinin kaçınca elemanı olduğunu bulan programı yazınız.

Uygulama-5: Girilen bir dizinin elemanları toplamını bulan programı yazınız.

Deney No: 5

Deney Adı: Sembolik İşlemler ve Polinomlar

1.Sembolik İşlemler

Sembolik işlemlerde kullanılacak değişkenlerin önceden bildirilmesi gerekir. Değişkenleri bildirmek için “syms” komutu kullanılır.

```
>> syms x,a,b,... % fonksiyon simgelerinin bildirilmesi
```

a) Belirsiz İntegral

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

Örnek-1: $\int \sin 2tx \cdot \cos 2x dx$ belirsiz integralin programını yazınız.

```
syms t x
f=sin(2*t*x)*cos(2*x);
sonuc=int(f)
pretty(sonuc)
```

b) Belirli İntegral

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

```
syms t x
f=sin(2*x)
sonuc=int(f,x,0,pi/2)
```

c) 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)
```

d) 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)
```

e) Denklem Sistemlerinin Çözümü

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

Örnek-5: $x+2y=1$, $x-y=2$ denklem sisteminin çözümü kümesini bulunuz.

```
[x,y]=solve('x+2*y=1','x-y=2')
```

f) Polinom Değerini Bulma

$P(x)$ polinomunun $x=k$ için alacağı değer “polyval” fonksiyonu ile bulunur. Polinomun kökleri ise “roots” komutuyla bulunur.

Örnek-6: $P(x)=3x^3+7x^2+5x+20$ polinomunun $x=5$ için değerini hesaplayınız.

```
P=[3 7 5 20];
polyval(P,5)
```

Örnek-7: $P(x)=3x^3+7x^2+5x+20$ polinomunun köklerini bulunuz.

```
P=[3 7 5 20];
kokler=roots(P)
```

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.

Grafik çizim komutları:

plot: Grafik çizimini sağlayan komuttur.

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	

hold on: Ekranda aynı anda birden fazla grafik görmeyi sağlar.

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.

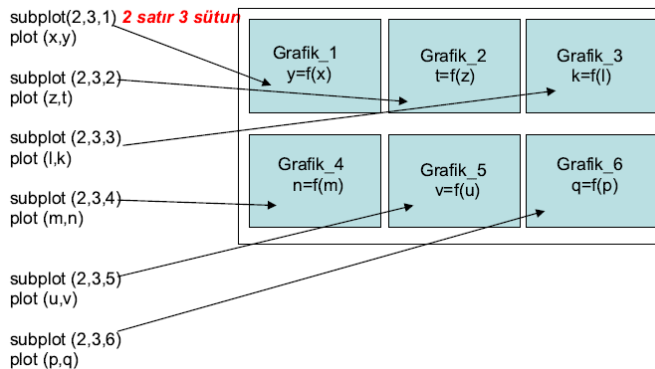


figure: Grafikleri farklı grafik pencerelerinde oluşturmak için “figure” fonksiyonu kullanılır.

Kullanım şekli;

figure(n): n pencerenin numarasıdır

Örnek-8: $P(t)=3t^3+7t^2+5t+20$ 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-9: 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--')
```

UYGULAMALAR

Uygulama-1: s^3+3s-4 polinomunun köklerini ve türevini bulan programını yazınız.

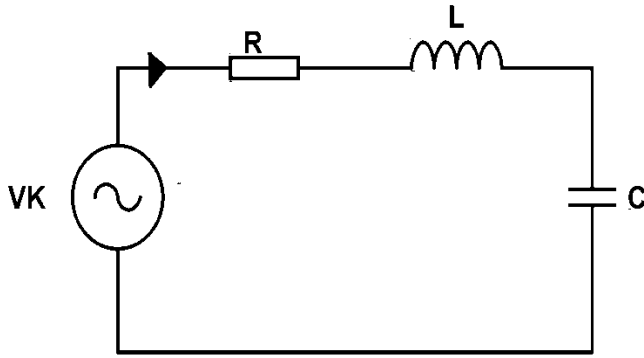
Uygulama-2: Ekranı 2 parçaya bölüp $\pi/2-\pi/2$ aralığında sin ve cos fonksiyonlarının değişimini çizdiriniz.

Uygulama-3: Aynı ekran üzerinde 0- π aralığında sinus ve cosinus fonksiyonlarının değişimini çizdiren programı yazınız.

Uygulama-4: Seri RC devresinde V_c geriliminin 0-10ms arasında değişimini çizdiriniz.

($R=100$; $C=50\mu F$; $V_k=5V$; $V_c= V_k (1-e^{-t/\tau})$; $\tau=RC$)

Uygulama-5:



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: 7

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** - FonksiyonAdi 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
```

UYGULAMALAR

Uygulama-1: Kendisine gönderilen iki polinomun çarpımını ve türevlerinin çarpımını çıkış olarak geri gönderen bir m-function yazınız.

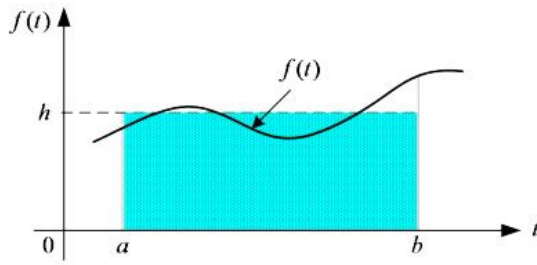
Uygulama-2: Kendisine gönderilen 2 dizinin toplam uzunluğunu bularak geri gönderen bir m-function oluşturunuz.

Uygulama-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: 8

Deney Adı: Sayısal İntegral Yöntemi

Sayısal integral, integrali alınacak fonksiyonun grafiği çizildiğinde grafiğin altında kalan alanın yaklaşık olarak hesaplanması prensibine dayanır.

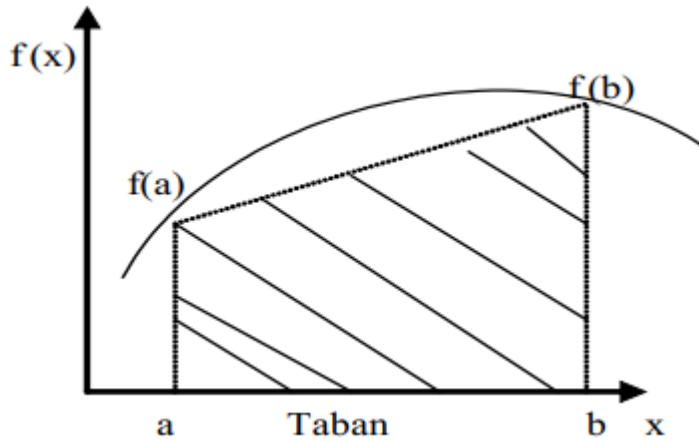


$$\text{Alan} = \int_a^b f(x) \cdot dx$$

Alan; $[a,b]$ aralığında $f(x)$ eğrisi ile x eksenini arasında kalan yüzeyin alanıdır. $f(x)$ fonksiyonunun integralinin alınması zor veya alınamıyorsa veya fonksiyon bilinmiyorsa sadece noktalar mevcut ise sayısal analiz yöntemleri belirli integralin bulunması için kullanılabilir.

1) Yamuk(Trapozoidal) Yöntemi;

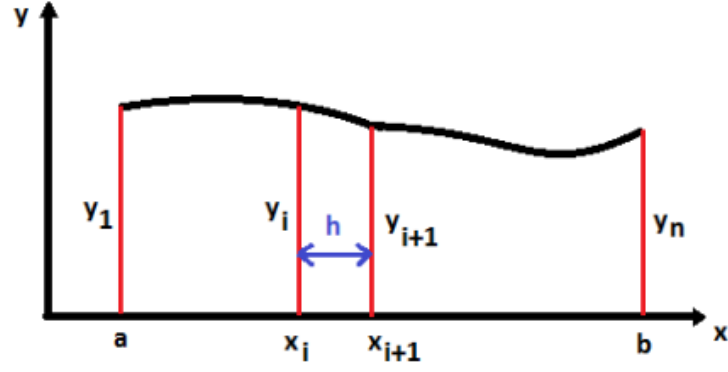
Fonksiyonun integrali, yamuğun alanına eşittir.



Bir yamuğun alanı;

$$\text{İntegral} = (b-a) \cdot \frac{f(b) + f(a)}{2}$$

Trapez yönteminde; şekilde görüldüğü gibi a ve b olarak sınırları belirlenen x eksenindeki aralığın h olarak adlandırılan eşit uzunlukta parçalara bölünmesi gerekir. Parçaların adedi ne kadar fazla olursa (h ne kadar küçük olursa) elde edilecek alan hesabının doğruluğu da o kadar yüksek olur.



$$Alan \cong h \left(\frac{y_1 + y_2}{2} \right) + h \left(\frac{y_2 + y_3}{2} \right) + h \left(\frac{y_3 + y_4}{2} \right) + \dots + h \left(\frac{y_{n-1} + y_n}{2} \right)$$

$$Alan \cong \frac{h}{2} [y_1 + 2(y_2 + y_3 + y_4 + \dots + y_{n-1}) + y_n]$$

Örnek-1: $y(x)=e^{-x^2}$ fonksiyonunun $0 < x < 1$ aralığı için $h=0.25$ kullanarak fonksiyona ait değer hesaplamaları aşağıdaki tabloda görülmektedir. Fonksiyonun integralini trapez yöntemi ile hesaplayınız.

n	x	$y(x)=e^{-x^2}$
1	0	1.000
2	0.25	0.9394
3	0.5	0.7788
4	0.75	0.5698
5	1	0.3679

$$Alan \cong \frac{0.25}{2} [1 + 2(0.9394 + 0.7788 + 0.5698) + 0.3679] = 0.7430$$

$$Alan \cong \frac{h}{2} [y_1 + 2(y_2 + y_3 + y_4 + \dots + y_{n-1}) + y_n]$$

Trapez komutu ile integral hesabı:

trapz(x,f(x))

x: İntegralin sınırları içerisinde h aralığına göre oluşan vektördür.

f(x): İntegrali hesaplanacak fonksiyondur.

Örnek-2: Örnek 1'i Matlab programı ile yazalım.

```
>>a=0; b=1; h=0.25;
```

```
>>alan=trapz(x,exp(-x.^2))
```

```
alan=
```

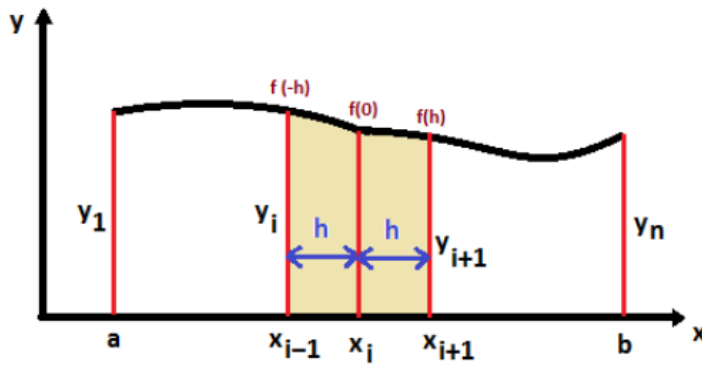
```
0.7430
```

2) Simpson Yöntemi

Bu yöntem, Trapez yöntemine göre alan hesabı daha sık aralıklarla yapılır. Trapez yöntemine göre daha doğru sonuçlar verir. Simpson yöntemi daha doğru integral hesabı için, noktaları birleştirmek amacıyla daha yüksek dereceli polinomlar kullanmaktır.

2.a. 1/3 Simpson Kuralı

Her üç noktadan 2. Dereceden bir polinom geçirilerek altındaki alanın yaklaşık olarak hesaplanması ilkesine dayanır. Bu yöntem, 2. Dereceden yaklaşımla yapıldığından dolayı en az 3 noktaya ihtiyaç vardır.



$$Alan \cong \frac{h}{3} [y_1 + 4(y_2 + y_4 + \dots + y_{n-1}) + 2(y_3 + y_5 + \dots + y_{n-2}) + y_n]$$

Simpson yönteminde, yukarıda şekilde görüldüğü gibi a ve b olarak sınırları belirlenen x eksenindeki aralığın h olarak adlandırılan eşit uzunlukta fakat çift sayıdan oluşan n adet parçalara bölünmesi gerekir. Parçaların adedi ne kadar fazla olursa (h ne kadar küçük olursa) elde edilecek alan hesabının doğruluğu da o kadar yüksek olur. Trapez yöntemine göre daha doğru sonuçlar verir.

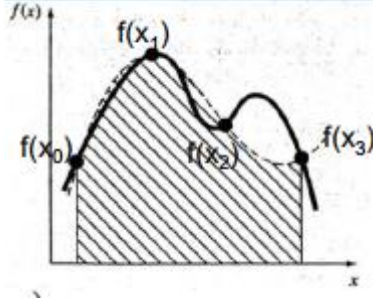
Örnek-3: Örnek-1'i Simpson 1/3 yöntemiyle hesaplayınız.

$$Alan \cong \frac{h}{3} [y_1 + 4(y_2 + y_4) + 2(y_3) + y_5]$$

$$Alan \cong \frac{0.25}{3} [1 + 4(0.9394 + 0.5698) + 2(0.7788) + 0.3679] = 0.7469$$

2.b. 3/8 Simpson Kuralı

(a) ve f(b) noktaları arasında eşit aralıklı iki nokta var ise, bu dört nokta üçüncü dereceden (kübik) bir polinom ile birleştirilebilir.



$$I \cong \frac{3h}{8} [f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)]$$

Örnek-4: $f(x) = 0.2 + 25x - 200x^2 + 675x^3 - 900x^4 + 400x^5$ fonksiyonunun $a=0$ 'dan $b=0.8$ 'e kadar

a) Simpson 1/3 kuralıyla,

b) Simpson 3/8 kuralıyla sayısal integralini hesaplayınız.

a) $f(0)=0.2$, $f(0.4)=2.456$, $f(0.8)=0.232$

$$I = 0.8 \frac{(0.2 + 4 * 2.456 + 0.232)}{6} = 1.367467$$

b) $f(0)=0.2$, $f(0.26674)=1.432724$, $f(0.5333)=3.487177$, $f(0.8)=0.232$

$$I = 0.8 \frac{(0.2 + 3 * 1.432724 + 3 * 3.487177 + 0.232)}{8} = 1.51917$$

UYGULAMALAR

Uygulama-1:

x	-1	-0.5	0	0.5	1
f(x)	-1	-0.5	0	1	2

Yandaki tabloda verilen $\int_{-1}^1 f(x)dx$ integralinin sayısal değerini Yamuk yöntemi, Simpson (1/8 ve 3/8) yöntemlerini kullanarak hesaplayınız.

Uygulama-2:

x	1	1.25	1.5	1.75	2
f(x)	6	7.0625	8.25	9.5625	11

Yandaki tabloda verilen $\int_1^2 f(x)dx$ integralinin sayısal değerini Yamuk yöntemi, Simpson (1/8 ve 3/8) yöntemlerini kullanarak hesaplayınız.

Uygulama-3: $\int_1^2 (x^2 + 2x + 3)dx$ integralinin sayısal değerini Simpson 3/8 yöntemini kullanarak 7 nokta için elde ediniz. Bu yönteme ilişkin matlab kodunu m-fonksiyon şeklinde yazınız.

Deney No: 9

Deney Adı: Newton-Raphson Yöntemi

Genel olarak $\nabla f(x) \neq 0$ gerek şartını sağlamak doğrusal olmayan ifadelerde oldukça zordur ve bu sebeple çözümler zor olabilir. Newton-Raphson yöntemi, doğrusal olmayan denklemlerin çözümü için iteratif(adım-adım) bir yaklaşım sunmaktadır. Aşağıdaki denklemi ele alalım:

$$f_i(x)=0, \quad i=1,2,\dots,m$$

x_0 verilmiş bir nokta olsun. **1. Mertebe Taylor açılımından:**

$$f(x) = f(x_0) + f'(x_0)(x_1 - x_0) = 0$$

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

Genel ifade;

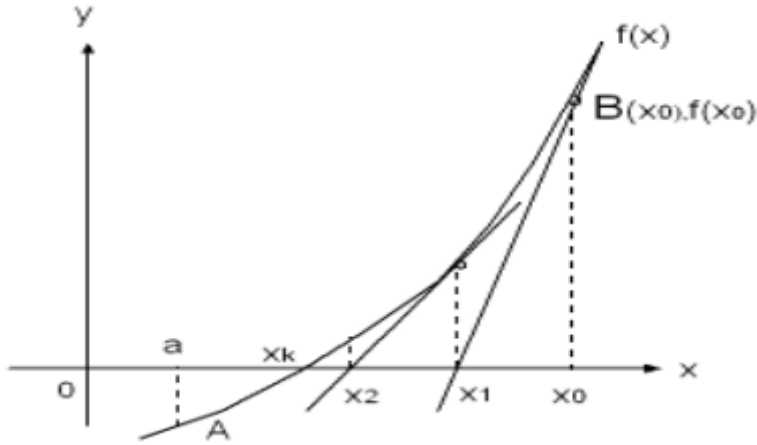
$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

Yukarıdaki ifade kullanılarak bir fonksiyonun kökü, yinelemeli yakınsama ile bulunmaya çalışılır. Bu ifadeyi aşağıdaki gibi yazmak da mümkündür.

$$f'(x_k) = \frac{f(x_k)}{x_k - x_{k+1}}$$

Buna göre, x_{k+1} noktası $f(x_k)$ noktasındaki eğiminden bulunacaktır. Burada $\tan \theta = f'(x_{k+1})$ 'dır. Bu durum aşağıdaki şekilden de incelenebilmektedir.

Fonksiyonun optimum noktasını bulmak için ise önce fonksiyonun türevi alınır ve türevi alınmış fonksiyona yukarıdaki işlemler uygulanır. Yakınsama her zaman mümkün olmayabilir. Aşağıdaki şekilde görüldüğü gibi başlangıç çözümü olarak a alındığında çözümden uzaklaşılacaktır. Genel olarak, yakınsama sağlanana kadar birçok başlangıç noktası seçmek gerekebilmektedir.



Newton-Raphson yöntemi Teğetler Yöntemi olarak da bilinir. Her bir noktanın teğetleriyle köke yaklaşılr.

Örnek-1: Aşağıdaki fonksiyonun durağan(optimum) noktalarını Newton-Raphson yöntemi ile bulunuz.

$$g(x)=(3x-2)^2(2x-3)^2$$

$$f(x)=g'(x)=144x^3-468x^2+482x-156=0$$

Sadeleştirme yapılırsa;

$$f(x)=g'(x)=72x^3-243x^2+241x-78=0$$

Newton-Raphson yöntemi için $f(x)$ fonksiyonunun türevini alırlz ve x_{k+1} noktalarını yinelemeli olarak elde ederiz.

$$f'(x)=g''(x)=216x^2-468x+241=0$$

$$x_{k+1} = x_k - \frac{f(x)}{f'(x)} = x_k - \frac{g'(x)}{g''(x)} = x_k - \frac{72x^3-234x^2+241x-78}{216x^2-468x+241}$$

$x_0=10$ noktasından başlayarak elde edilen yeni noktalar ve yaklaşım çözüm değeri aşağıdaki tabloda verilmiştir.

k	x_k	$f(x)$	$f'(x)$	$f(x)/f'(x)$	x_{k+1}
0	10,000000	50932	17161	2,9678923	7,032108
1	7,032108	15082,7	7631,29	1,9764288	5,055679
2	5,055679	4463,431	3395,878	1,3143672	3,741312
3	3,741312	1318,807	1513,507	0,871358	2,869954
4	2,869954	388,277	676,9746	0,5735474	2,296406
5	2,296406	113,3633	305,3539	0,371252	1,925154
6	1,925154	32,43006	140,5711	0,2307022	1,694452
7	1,694452	8,793732	68,16869	0,1289996	1,565453
8	1,565453	2,042065	37,70681	0,0541564	1,511296
9	1,511296	0,293991	27,06086	0,0108641	1,500432
10	1,500432	0,010818	25,07781	0,0004314	1,500001