



**Yıldız Teknik Üniversitesi  
Elektrik-Elektronik Fakültesi  
Bilgisayar Mühendisliği Bölümü**

**BLM1022**

**Sayısal Analiz**

**Gr: 1**

**Prof. Dr. Banu Diri**

**Dönem Projesi**

**Batuhan ODÇIKIN**

**22011093**

## İçindekiler

Ön bilgi .....	3
Ana Menü .....	3
Desteklenen Fonksiyonlar .....	4
Polimon .....	4
Üstel .....	4
Trigonometrik .....	4
Ters Trigonometrik .....	4
Matris Girişi .....	5
Bisection Yöntemi .....	6
Parametreler .....	6
Örnek .....	6
Regula-Falsi Yöntemi .....	7
Newton-Rapshon Yöntemi .....	7
Parametreler .....	7
Örnek .....	7
NxN'lik bir Matrisin Tersi .....	8
Parametreler .....	8
Örnek .....	8
Gauss Eliminasyon Yöntemi .....	9
Parametreler .....	9
Örnek .....	9
Gauss Seidel Yöntemi .....	10
Parametreler .....	10
Örnek .....	10
Sayısal Türev .....	11
Parametreler .....	11
Örnek .....	11
Simpson Yöntemi .....	12
Parametreler .....	12
Örnek .....	12
Trapez Yöntemi .....	13
Parametreler .....	13
Örnek .....	13
Değişken Dönüşümsüz Gregory-Newton Enterpolasyonu .....	14
Parametreler .....	14

## Ön bilgi

Program, 10 tane belirli işlemi yerine getirebilmek için tasarlanmıştır. Bu işlemler sırasıyla şöyledir:

1. Bisection yöntemi
2. Regula-Falsi yöntemi
3. Newton-Rapshon yöntemi
4. NxN'lik bir matrisin tersi
5. Gauss eliminasyon yöntemi
6. Gauss-Seidel yöntemi
7. Sayısal Türev
8. Simpson yöntemi
9. Trapez yöntemi
10. Değişken dönüşümsüz Gregory-Newton enterpolasyon

1	2	3	4	5	6	7	8	9	10
1	1	1	1	1	1	1	1	1	1

## Ana Menü

Kod çalıştırıldıktan sonra 1 ile 10 arasında bir sayı girilerek istenilen yöntem seçilebilir.

Program -1 girilene kadar çalışmaya devam eder.

```

      Methot      |Durum
1-Bisection Method |+
2-Regula-Falsi Method |+
3-Newton-Rapshon Method |+
4-Reverse of Matrix |+
5-Gauss Elimination |+
6-Gauss Seidal Methot |+
7-Numeric Derivative |+
8-Simpson Methot |+
9-Trapez Methot |+
10-Gergory Newton Methot|+

!!!!!!===== DIKKAT =====!!!!!!
!
- x'in katsayılarını (5*x) şeklinde giriniz
- İşlem önceliklerini parantez ile belirtmeyi unutmayınız

Press [-1] for exit

Enter the methot: 

```

## Desteklenen Fonksiyonlar

Kök bulma yöntemleri (1, 2, 3), sayısal türev ve integral yöntemleri (7, 8, 9) ve enterpolasyon yöntemleri (10) için ilk istenilen parametre fonksiyondur. Bu fonksiyon sırasıyla polinom, üstel, logaritmik, trigonometrik ve ters trigonometrik fonksiyon tiplerini barındıracak şekilde ayarlanabilir. Yalnızca fonksiyon girilirken işlem önceliğinin parantezler yardımıyla gösterilmesi gerekmektedir. Aynı zamanda programın düzgün çalışabilmesi adına  $x$ 'in katsayısı da  $(5*x)$  şeklinde çarpma işlemiyle ifade edilmelidir.

### Polimon

$$a*x^{(b)} + c *x^d...$$

$$=axb + cx d + \dots$$

a:kat sayısı

b:üs

### Üstel

$$b*x^a$$

$$b*x^{(a)}$$

$$(b)*x^a='$$

b =  $x$ 'in kat sayısı

a =  $x$ 'in üstü

### Trigonometrik

$$\sin(a*x), \cos(a*x), \tan(a*x), \cot(a*x), \sec(a*x), \operatorname{cosec}(a*x)$$

### Ters Trigonometrik

$$\operatorname{asin}(a*x), \operatorname{acos}(a*x), \operatorname{atan}(a*x)$$

## Matris Girişi

Matris girişi yapmak için önce boyutu (N) girilir, sonrasında Sıra sıra (i,j): formatında matrisin elemanlarının girişi yapılır.

```
Enter the methot: 4
Reverso of Matrix
Enter the dimensions of the matrix (N): 3
Enter the matrix
(0,0):1
(0,1):2
(0,2):3
(1,0):4
(1,1):5
(1,2):6
(2,0):7
(2,1):2
(2,2):9
Original Matrix:

1.00 2.00 3.00
4.00 5.00 6.00
7.00 2.00 9.00
```

# Bisection Yöntemi

Fonksiyon, başlangıç ve bitiş değerleri girilir. Hata  $|\text{son değer} - \text{bir önceki değer}| < \text{ERR}$  ile hesaplanır. ERR değeri 0.00002 olarak minimum değerde define edilmiştir

## Parametreler

**Fonksiyon girişi**

**Range values: base & top** (başlangıç ve bitiş değerleri)

## Örnek

Fonksiyon:  $x^3 + (2 \cdot x) + 2$

Base value: -20

Top value: 25

ERR: 0.00002

Stop criteria:  $|\text{son değer} - \text{bir önceki değer}| < \text{ERR}$

```
Enter the method: 1
Bisection Method
Enter the function: x^3+(2*x)+2
Enter the range values
Base Value: -20
Top Value: 25
err: 22.500000
iteration: 1
(a,x,b)
(-20.000000,2.500000,25.000000)
(ax,xb)
(-1,1)
=====
err: 5.500000
iteration: 2
(a,x,b)
(-20.000000,-8.750000,2.500000)
(ax,xb)
(1,-1)
=====
err: 1.250000
iteration: 3
(a,x,b)
(-8.750000,-3.125000,2.500000)
(ax,xb)
(1,-1)
=====
err: 0.312500
iteration: 4
(a,x,b)
(-3.125000,-0.312500,2.500000)
(ax,xb)
(1,-1)
=====
err: 0.062500
iteration: 5
(a,x,b)
(-0.312500,1.093750,2.500000)
(ax,xb)
(-1,1)
=====
err: 0.015625
iteration: 6
(a,x,b)
(-0.312500,0.390625,1.093750)
(ax,xb)
(-1,1)
=====
err: 0.000000
iteration: 7
(a,x,b)
(-0.312500,0.039062,0.390625)
(ax,xb)
(-1,1)
=====
The root is: 0.039062
```

# Regula-Falsi Yöntemi

Regula yöntemin doğru çalışmamaktadır

# Newton-Rapshon Yöntemi

Fonksiyon ve başlangıç değerini ister, hata  $|\text{son değer} - \text{bir önceki değer}| < \text{ERR}$  ile bulunur. ERR değeri ise en düşük değer olan 0.00002 olarak define edilmiştir.

## Parametreler

**Fonksiyon**

**Başlangıç değeri**

## Örnek

Fonskyion:  $x^3 + (2 \cdot x) + 2$

Başlangıç değeri: 3

ERR: 0.00002

Stop criteria:  $|\text{son değer} - \text{bir önceki değer}| < \text{ERR}$

```
Enter the methot: 3
Newton Rapshon Methot
Enter the first value:3
1.iteration||x:-0.998644      err:0.998644
2.iteration||x:-0.799913     err:0.198730
3.iteration||x:-0.771465     err:0.028449
4.iteration||x:-0.770918     err:0.000547
5.iteration||x:-0.770917     err:0.000001
Root:-0.770917
```

# NxN'lik bir Matrisin Tersİ

## Parametreler

N: Matrisin boyutu

Matris : Ters çevrilmek istenen matris

## Örnek

N: 3

Matris: {(1,2,3), (4,5,6), (7,2,9)} (En son matris girilen matrisin tersi, diğer matrisler ters çevirme adımlarını göstermektedir.)

```
Enter the method: 4
Reverso of Matrix
Enter the dimensions of the matrix (N): 3
Enter the matrix
(0,0):1
(0,1):2
(0,2):3
(1,0):4
(1,1):5
(1,2):6
(2,0):7
(2,1):2
(2,2):9
Original Matrix:

1.00 2.00 3.00
4.00 5.00 6.00
7.00 2.00 9.00

Reversing Matrix...

Memory Allocated.
Matrix:
1.00 2.00 3.00
0.00 5.00 6.00
7.00 2.00 9.00

Inverse:
1.00 0.00 0.00
-4.00 1.00 0.00
0.00 0.00 1.00

Matrix:
1.00 2.00 3.00
0.00 -3.00 6.00
7.00 2.00 9.00

Inverse:
1.00 0.00 0.00
-4.00 1.00 0.00
0.00 0.00 1.00

Matrix:
1.00 0.00 0.00
0.00 -3.00 0.00
0.00 0.00 12.00

Inverse:
-0.92 0.33 0.08
0.50 -1.00 0.00
9.00 -4.00 1.00

Matrix:
1.00 0.00 0.00
0.00 -3.00 0.00
0.00 0.00 12.00

Inverse:
-0.92 0.33 0.08
0.50 -1.00 0.50
9.00 -4.00 1.00

Matrix:
1.00 0.00 0.00
0.00 -3.00 0.00
0.00 0.00 1.00

Inverse Matrix:
-0.92 0.33 0.08
-0.17 0.33 -0.17
0.75 -0.33 0.08
```



# Gauss Eliminasyon Yöntemi

## Parametreler

N: Matrisin boyutu

Matris: İşlenmek istenen matris

## Örnek

N: 4

Matris:  $\{(1,2,3,13), (4,5,6,14), (7,2,9,15), (10,11,12,16)\}$  (En son matris girilen matrisin tersi, diğer matrisler ters çevirme adımlarını göstermektedir.)

```
Enter the method: 5
Gauss Elimination Method
Enter the dimensions of the matrix (N): 4
Enter the matrix
(0,0):1
(0,1):2
(0,2):3
(0,3):13
(1,0):4
(1,1):5
(1,2):6
(1,3):14
(2,0):7
(2,1):2
(2,2):9
(2,3):15
(3,0):10
(3,1):11
(3,2):12
(3,3):16
Original Matrix
1.00 2.00 3.00 13.00
4.00 5.00 6.00 14.00
7.00 2.00 9.00 15.00
10.00 11.00 12.00 16.00

1.00 2.00 3.00 13.00
0.00 -3.00 -6.00 -38.00
0.00 0.00 12.00 76.00
0.00 0.00 0.00 0.00
```

# Gauss Seidel Yöntemi

## Parametreler

N: Matrisin boyutu

Katsayı matrisi: Çözölmek istenen matris

Sonuç Matrisi: Sonuç matrisi

## Örnek

N:4

Katsayı Matrisi: {(45,2,3), (-3,22,2), (5,1,20)}

Sonuç Matrisi: {58,47,67}

```
Enter the method: 6
Gauss Seidel Method
Enter the dimensions of the matrix (N): 3
Enter the matrix
(0,0):45
(0,1):2
(0,2):3
(1,0):-3
(1,1):22
(1,2):2
(2,0):5
(2,1):1
(2,2):20
Original Matrix
45.00 2.00 3.00
-3.00 22.00 2.00
5.00 1.00 20.00

Enter the [C] matrix
(0,0):58
(0,1):47
(0,2):67

0.iteration
v0: 1.288889 || err:1.288889
v1: 2.312121 || err:2.312121
v2: 2.912172 || err:2.912172

1.iteration
v0: 0.991983 || err:0.296906
v1: 2.006891 || err:0.305230
v2: 3.001660 || err:0.089488

2.iteration
v0: 0.999583 || err:0.007600
v1: 1.999792 || err:0.007099
v2: 3.000114 || err:0.001545

3.iteration
v0: 1.000002 || err:0.000419
v1: 1.999990 || err:0.000198
v2: 3.000000 || err:0.000114

4.iteration
v0: 1.000000 || err:0.000001
v1: 2.000000 || err:0.000010
v2: 3.000000 || err:0.000000
```

# Sayısal Türev

## Parametreler

Fonksiyon

Türev alma yöntemi (ileri, geri veya merkezi)

X: Türev alınmak istenen nokta

H: Türevin alınacağı fark

## Örnek

Fonksiyon:  $x^3 + (2 \cdot x) + 2$

Türev alma yöntemi : 3 (merkezi)

X: 2

H:0.0001

```
Enter the method: 7
Numeric Derivative
Enter the function: x^3+(2*x)+2
Please choose the way:
1-ileri fark
2-geri fark
3-merkezi fark
Selection: 3
Enter the x (6D):2
Enter the H (6D):0.0001,
Result of numeric derivative is (x):13.998402
```

# Simpson Yöntemi

## Parametreler

Fonksiyon

[a,b]: Aralık Üst ve Alt Değerleri

n: Adım Sayısı

## Örnek

Fonksiyon:  $x^3+(2*x)+2$

[a,b]: [1,5]

n: 10

```
Enter the method: 8
Trapez Method
Enter the function: x^3+(2*x)+2
Enter the range values(a,b):
1
5
Enter the number of steps(n):10

Trapez Method Integral:
Range [1.00,5.00], n = 10, step_val= 0.400000
  x    ||    f(x)
f(x00) = 3.000000    | x0 = 1.000000
f(x01) = 5.544000    | x1 = 1.400000 *4
f(x02) = 9.431999    | x2 = 1.800000 *2
f(x03) = 15.048000    | x3 = 2.200000 *4
f(x04) = 22.775997    | x4 = 2.600000 *2
f(x05) = 33.000000    | x5 = 3.000000 *4
f(x06) = 46.104004    | x6 = 3.400000 *2
f(x07) = 62.471996    | x7 = 3.800000 *4
f(x08) = 82.487991    | x8 = 4.200000 *2
f(x09) = 106.535988    | x9 = 4.600000
Integral of f(x) at range [1.00,5.00] = 123.180801
```

# Trapez Yöntemi

## Parametreler

Fonksiyon

[a,b]: Aralık Üst ve Alt Değerleri

n: Adım Sayısı

## Örnek

Fonksiyon:  $x^3+(2*x)+2$

[a,b]: [1,5]

n: 10

```
Enter the method: 9
Trapez Method
Enter the function: x^3+(2*x)+2
Enter the range values(a,b):
1
5
Enter the number of steps(n):10

Trapez Method Integral:
Range [1.00,5.00], n = 10, step_val= 0.400000

f(x00) = 3.000000      | x0 = 1.000000
f(x01) = 5.544000      | x1 = 1.400000
f(x02) = 9.431999      | x2 = 1.800000
f(x03) = 15.048000      | x3 = 2.200000
f(x04) = 22.775997      | x4 = 2.600000
f(x05) = 33.000000      | x5 = 3.000000
f(x06) = 46.104004      | x6 = 3.400000
f(x07) = 62.471996      | x7 = 3.800000
f(x08) = 82.487991      | x8 = 4.200000
f(x09) = 106.535988     | x9 = 4.600000
Integral of f(x) at range [1.00,5.00] = 126.959984
```

# Değişken Dönüşümsüz Gregory-Newton Entropolasyonu

## Parametreler

Değer sayısı

Adım sayısı

f(x):Değerler

x: oluşturulan fonksiyonda bulunmak istenen x değeri

## Örnek

Değer sayısı:5

Adım sayısı:4

f(x):1, 3, 5, 8, 10

x: 6

```
Enter the method: 10
Gregory Newton Interpolation Method
Enter number of values: 5
Enter maximum step number: 4
f(0): 1
f(1): 3
f(2): 5
f(3): 8
f(4): 10
Data get
1.00 2.00 0.00 1.00 0.00
3.00 2.00 1.00 -2.00
5.00 3.00 -1.00
8.00 2.00
10.00

Factor array: 1.000000 2.000000 0.000000 1.000000

The function:
1.00 /0! + 2.00 x/1! + 0.00 x(x-1)/2! + 1.00 x(x-1)(x-2)/3!
Enter the x for f(x):6
x: 6.000000
sum: 1.000000 = 1.000000*1.000000/0.000000
(1,0)-6.000000,0.000000,0.000000
sum: 13.000000 = 2.000000*6.000000/0.000000
(2,0)-6.000000,0.000000,0.000000
(2,1)-30.000000,0.000000,0.000000
sum: 13.000000 = 0.000000*30.000000/0.000000
(3,0)-6.000000,0.000000,0.000000
(3,1)-30.000000,0.000000,0.000000
(3,2)-120.000000,0.000000,0.000000
sum: 53.000000 = 1.000000*120.000000/0.000000
Result: 53.000000
```