



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

**BLM1022
Sayısal Analiz
Gr: 2
Öğr. Gör. Dr. Ahmet ELBİR
Dönem Projesi**

İsim: ENES EMIN ÖZBİLGİN

No: 23011914

E-posta: eneseminozbilgin@gmail.com

İçindekiler

Ön Bilgi	4
Ana Menü	5
Desteklenen Fonksiyonlar	6
Polinom	6
Üstel	6
Logaritmik	6
Trigonometrik	6
Ters Trigonometrik	7
Örnekler	7
Matris Girişi	10
Örnek	10
Bisection Yöntemi	11
Parametreler	11
Örnek	11
Regula-Falsi Yöntemi	12
Parametreler	12
Örnek	12
Newton-Raphson Yöntemi	13
Parametreler	13
Örnek	13
NxN'lik Bir Matrisin Tersi	14
Parametreler	14
Örnek	14
Gauss Eliminasyon Yöntemi	15
Parametreler	15
Örnek	15
Gauss-Seidel Yöntemi	16
Parametreler	16
Örnek	16
Sayısal Türev	17
Parametreler	17
Örnek	17
Simpson Yöntemi	18
Parametreler	18
Örnek	18

Trapez Yöntemi	19
Parametreler	19
Örnek	19
Değişken Dönüşümsüz Gregory-Newton Enterpolasyonu	20
Parametreler	20
Örnekler	20

Ön Bilgi

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

1. Bisection yöntemi → 1
2. Regula-Falsi yöntemi → 1
3. Newton-Raphson yöntemi → 1
4. $N \times N$ 'lik bir matrisin tersi → 1
5. Gauss eliminasyon yöntemi → 1
6. Gauss-Seidel yöntemi → 1
7. Sayısal Türev → 1
8. Simpson yöntemi → 1
9. Trapez yöntemi → 1
10. Değişken dönüşümsüz Gregory-Newton enterpolasyonu → 1

Yöntemlerin hepsi çalışmaktadır.

Ana Menü

Çalıştırılmak istenilen işlem program çalıştırıldıktan sonra numarası girilip gereken parametrelerin verilmesiyle çalışır. Ana menüde '0' girdisi verilene kadar program çalışmaya devam eder.

```
Quit: 0
Bisection: 1
Regula-Falsi: 2
Newton Raphson: 3
Inverse Matrix: 4
Gauss Elimination: 5
Gauss-Seidel: 6
Numerical Differentiation: 7
Simpson's Rule: 8
Trapezoidal Rule: 9
Gregory-Newton: 10
Choice:
```

Desteklenen Fonksiyonlar

Kök bulma yöntemleri (1, 2, 3), sayısal türev ve integral yöntemleri (7, 8, 9) ve interpolasyon yöntemleri için ilk istenilen parametre fonksiyondur. Bu fonksiyon sırasıyla polinom, üstel, logaritmik, trigonometrik ve ters trigonometrik fonksiyon tiplerini barındıracak şekilde ayarlanabilir. Bu tiplerin parametreleri şöyledir:

Polinom

$$x_{coef} \times x^{x_{exp}}$$

x_{coef} : x'in katsayısı

x_{exp} : x'in üstü

Üstel

$$fn_{coef} \times (base^{(x_{coef} \times x^{x_{exp}})})^{fn_{exp}}$$

x_{coef} : x'in katsayısı

x_{exp} : x'in üstü

fn_{coef} : Fonksiyonun katsayısı

fn_{exp} : Fonksiyonun üstü

$base$: Üstel ifadenin tabanı

Logaritmik

$$fn_{coef} \times (\log_{base}(x_{coef} \times x^{x_{exp}}))^{fn_{exp}}$$

x_{coef} : x'in katsayısı

x_{exp} : x'in üstü

fn_{coef} : Fonksiyonun katsayısı

fn_{exp} : Fonksiyonun üstü

$base$: Logaritmanın tabanı

Trigonometrik

$$fn_{coef} \times trig_{fn}(x_{coef} \times x^{x_{exp}})^{fn_{exp}}$$

$$trig_{fn}: \begin{cases} \sin, & 0 \\ \cos, & 1 \\ \tan, & 2 \\ \cot, & 3 \end{cases}$$

x_{coef} : x'in katsayısı

x_{exp} : x'in üstü

fn_{coef} : Fonksiyonun katsayısı

fn_{exp} : Fonksiyonun üstü

Ters Trigonometrik

$$fn_{coef} \times trig_{fn}(x_{coef} \times x^{x_{exp}})^{fn_{exp}}$$

$$trig_{fn}: \begin{cases} \arcsin, & 0 \\ \arccos, & 1 \\ \arctan, & 2 \\ \operatorname{arccot}, & 3 \end{cases}$$

x_{coef} : x'in katsayısı

x_{exp} : x'in üstü

fn_{coef} : Fonksiyonun katsayısı

fn_{exp} : Fonksiyonun üstü

Burada görüldüğü gibi olan fonksiyonlar destekleniyor.

Ayrıca buradaki fonksiyonlardan her biri birbiriyle çarpımı, bölümü, toplamı ve çıkarması da desteklenmektedir.

Dikkat edilmesi gereken hususlar:

1-Katsayı çarpımı ile Terimler arası çarpımlar farklıdır.

Örnek: $2 \cdot x^2 \times 2 \cdot (e^{(4 \cdot x^2)})^2$: bu bir polinom ile üstel ifadenin çarpımıdır. Katsayı çarpımında (\cdot) kullanılırken , Terimler arası çarpımda (\times) kullanılmıştır. Bu şekilde fonksiyonu yazdığınızdan emin olun.

2-Bölme işaretimiz (/) dir.

3- Trigonometrik fonksiyonlarımızda ve öbür ifadelerimizde fonksiyonun içine başka bir türden ifade girilemez.

Örnek: $\sin(x \cdot e^{(5x)})$, $x^{(\sin(\log_5(x^3)))}$, $\log_x(\sin(5x^2 + \sin 5x))$ tarzı ifadeler desteklenmemektedir.

4-Girilen fonksiyonlarda $2 \cdot x$ gibi ifadeler girilebilir, $2 \cdot x^1$ girilmesi zorunlu değildir.

5-Girilen fonksiyonlarda x^3 gibi ifadeler girilebilir, $1 \cdot x^3$ girilmesi zorunlu değildir.

6-Logaritmik fonksiyonda $\log_e(x)$ yerine $\ln(x)$ kullanılabilir.

7-Üstel fonksiyonda e harfi euler sayısına denktir. $2 \cdot (e^{(4 \cdot x^2)})^2$ tarzı ifadeler girilebilir.

8-epsilon değeri girilirken 0,001 gibi ondalık kısım girilirken ',' kullanılmasına dikkat edelim. compiler '.' tanımadığından 0.001 gibi bir sayıyı 0'a çevirecektir. Bu gibi durumlardan kaçınmak için nokta yerine virgöl kullandığımızdan emin olalım.

Matris Girişi

Matrisin tersi (4) ve lineer denklem çözümü yöntemleri (5, 6) için ilk istenilen parametre $N \times N$ 'lik bir kare matris için N değeridir. Bu değer girildikten sonra matrisin elemanları satır satır alınır.

Örnek

$$N = 3, \quad \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

```
N:
3
[0][0]:
1
[0][1]:
2
[0][2]:
3
[1][0]:
4
[1][1]:
5
[1][2]:
6
[2][0]:
7
[2][1]:
8
[2][2]:
9
Matrix: [
    1.000000    2.000000    3.000000
    4.000000    5.000000    6.000000
    7.000000    8.000000    9.000000
]
```


Bisection Yöntemi

Parametreler

Fonksiyon

start: Başlangıç değeri

end: Bitiş değeri

epsilon: Hata miktarı

Stopping criterion: Durma koşulu = $\begin{cases} f(x) \leq \epsilon, 1 \\ \frac{end - start}{2^n} \leq \epsilon, 2 \end{cases}$

Max iterations: Maksimum iterasyon sayısı

Örnek

Fonksiyon: $x^3 - 7x^2 + 14x - 6$

start: 0

end: 1

epsilon: 0.01

Stopping criterion: 2 (Durma koşulu = $\frac{end - start}{2^n} \leq \epsilon$)

Max iterations: 100

```
Function: 1.000000 * x ^ 3.000000 + -7.000000 * x ^ 2.000000 + 14.000000 * x ^ 1.000000 + -6.000000 * x ^ 0.000000

start:
0
end:
1
epsilon:
0.01
Stopping criterion:
f(x) <= epsilon: 1
(end - start) / 2^n <= epsilon: 2
Choice:
2
Max iterations:
100

start : +0.000000
end : +1.000000
mid : +0.500000
f(start) : -6.000000
f(end) : +2.000000
f(mid) : -0.625000
iteration : 1

start : +0.500000
end : +1.000000
mid : +0.750000
f(start) : -0.625000
f(end) : +2.000000
f(mid) : +0.984375
iteration : 2

start : +0.500000
end : +0.750000
mid : +0.625000
f(start) : -0.625000
f(end) : +0.984375
f(mid) : +0.259766
iteration : 3

start : +0.500000
end : +0.625000
mid : +0.562500
f(start) : -0.625000
f(end) : +0.259766
f(mid) : -0.161865
iteration : 4
Result: 0.562500
```

Regula falsi Yöntemi

Parametreler

Fonksiyon

start: Başlangıç değeri

end: Bitiş değeri

epsilon: Hata miktarı

Max iterations: Maksimum iterasyon sayısı

```
Fonksiyonu girin: 2*x^2 - 3*x - 5

REGULAFALSI YONTEMI ILE KOK BULMA
Koku icerisine alan ilk degeri girin: 0
Koku icerisine alan ikinci degeri girin: 3
epsilon degerini giriniz: 1
max iterasyon degerini giriniz: 100

start      : 0,000000
end        : 3,000000
mid        : 1,666667
F(start)   : -5,000000
F(end)     : 4,000000
F(mid)     : -4,444445
iteration: 1

start      : 1,666667
end        : 3,000000
mid        : 1,666667
F(start)   : -4,444445
F(end)     : 4,000000
F(mid)     : -4,444445
iteration: 2

start      : 2,368421
end        : 3,000000
mid        : 2,368421
F(start)   : -0,886426
F(end)     : 4,000000
F(mid)     : -0,886426
iteration: 3

start      : 2,482993
end        : 3,000000
mid        : 2,482993
F(start)   : -0,118469
F(end)     : 4,000000
F(mid)     : -0,118469
iteration: 4
```

```
start      : 2,497865
end        : 3,000000
mid        : 2,497865
F(start)   : -0,014936
F(end)     : 4,000000
F(mid)     : -0,014936
iteration: 5

start      : 2,499733
end        : 3,000000
mid        : 2,499733
F(start)   : -0,001869
F(end)     : 4,000000
F(mid)     : -0,001869
iteration: 6

start      : 2,499967
end        : 3,000000
mid        : 2,499967
F(start)   : -0,000234
F(end)     : 4,000000
F(mid)     : -0,000234
iteration: 7

Denklemin koku: 2,49997
```

Newton-Raphson Yöntemi

Parametreler

Fonksiyon

x_0 : x 'in başlangıç değeri

epsilon: Hata miktarı

Max iterations: Maksimum iterasyon sayısı

Örnek

Fonksiyon: $x^3 - 7x^2 + 14x - 6$

x_0 : 0

epsilon: 0.000001

Max iterations: 100

```
Function: 1.000000 * x ^ 3.000000 + -7.000000 * x ^ 2.000000 + 14.000000 * x ^ 1.000000 + -6.000000 * x ^ 0.000000
x0:
0
epsilon:
0.000001
Max iterations:
100

xn      : +0.000000
xn+1    : +0.428571
f(xn)   : -6.000000
f'(xn)  : +14.000000
iteration : 1

xn      : +0.428571
xn+1    : +0.569724
f(xn)   : -1.206997
f'(xn)  : +8.551020
iteration : 2

xn      : +0.569724
xn+1    : +0.585592
f(xn)   : -0.111039
f'(xn)  : +6.997622
iteration : 3

xn      : +0.585592
xn+1    : +0.585786
f(xn)   : -0.001328
f'(xn)  : +6.830466
iteration : 4

xn      : +0.585786
xn+1    : +0.585786
f(xn)   : -0.000000
f'(xn)  : +6.828427
iteration : 5
Result: 0.585786
```

Inverse matrix bulma

Bulmak istediğiniz matrisin boyutunu girin.
Ardından matrisin kendisini girin.

```
Enter the size of the square matrix: 3
```

```
Enter the elements of the matrix:
```

```
[0][0]: 1
```

```
[0][1]: 3
```

```
[0][2]: 4
```

```
[1][0]: 2
```

```
[1][1]: 3
```

```
[1][2]: 1
```

```
[2][0]: 3
```

```
[2][1]: 2
```

```
[2][2]: 1
```

```
Entered matrix:
```

```
| 1,0000 3,0000 4,0000 |  
| 2,0000 3,0000 1,0000 |  
| 3,0000 2,0000 1,0000 |
```

```
Inverse of the matrix:
```

```
| -0,0625 -0,3125 0,5625 |  
| -0,0625 0,6875 -0,4375 |  
| 0,3125 -0,4375 0,1875 |
```

Gauss Elimination

Çözmek istediğimiz eşitsizliklerin denklem sayısını gireriz.

Ardından Augmented matriximizi gireriz. ($2x + 4y + 5z = 12 \rightarrow 2 \ 4 \ 5 \ 12$ şeklinde gireriz).

GAUSS ELIMINATION METHODU İLE DENKLEM COZUMU

Enter the number of equations: 3

Enter the augmented matrix:

[0][0]: 2
[0][1]: 3
[0][2]: 4
[0][3]: 12
[1][0]: 6
[1][1]: 4
[1][2]: 7
[1][3]: 23
[2][0]: 1
[2][1]: 2
[2][2]: 1
[2][3]: 5

Solution:

x[1] = 1,333
x[2] = 1,067
x[3] = 1,533

Gauss Seidel

Çözümünü bulmak istediğiniz denklemlerin Augmented matrixini giriniz.

Örnek: $(2x + 4y + 5z = 12 \rightarrow 2 \ 4 \ 5 \ 12$ şeklinde gireriz).

Matrisimiz 3x3 olduğu varsayılır.

GAUSS SEIDEL YONTEMI ILE DENKLEM COZUMU

Matrixi giriniz:

[0][0]: 5

[0][1]: 2

[0][2]: 1

[0][3]: 12

[1][0]: 2

[1][1]: 3

[1][2]: 4

[1][3]: 20

[2][0]: -4

[2][1]: 5

[2][2]: 3

[2][3]: 15

epsilon degerini giriniz:

0,001

Count	x	y	z
1	2,4000	4,9200	0,1100
2	0,4100	3,2620	2,3485
3	0,6255	2,0913	3,1188
4	0,9397	1,8805	3,1198

Solution: $x = 0,9397$, $y = 1,8805$ and $z = 3,1198$

Numerical Differentiation

İleri fark , Geri fark , Orta nokta ve İkinci türevlerden birini seçeriz.
Fonksiyonumuzu gireriz.
H değerimizi gireriz.
Ardından türevini bulmak istediğimiz noktayı gireriz.

```
SAYISAL TUREV İLE TUREV BULMA
Please choose one -> forward diff (f),backward diff (b),central diff (c),2nd diff (d):f
Fonksiyonu girin: 2*x^3 - 4*x
lutfen h degerini giriniz: 0,01
lutfen turevini almak istediginiz x noktasini giriniz: 2

x = 2,000000 noktasindaki ileri turev : 19,9995
```

```
SAYISAL TUREV İLE TUREV BULMA
Please choose one -> forward diff (f),backward diff (b),central diff (c),2nd diff (d):b
Fonksiyonu girin: 2*x^3 - 4*x
lutfen h degerini giriniz: 0,01
lutfen turevini almak istediginiz x noktasini giriniz: 2

x = 2,000000 noktasindaki geri turev : 19,9996
```

Simpson's Rule

Hangi Simpson yöntemini kullanacağımızı seçeriz.

Fonksiyonu gireriz.

Başlangıç ve bitiş noktalarımızı gireriz.

```
SIMPSON'S RULE ILE INTEGRAL BULMA
```

```
lutfen bir tanesini secin:
```

```
simpson 1/3 kurali (0):
```

```
simpson 3/8 kurali (1):
```

```
0
```

```
Fonksiyonu girin: 3*x^2 + 2*x
```

```
lutfen baslangic degerini giriniz: 1
```

```
lutfen bitis degerini giriniz: 3
```

```
simpson 1/3 kuralina gore intergral = 34,0000
```

```
SIMPSON'S RULE ILE INTEGRAL BULMA
```

```
lutfen bir tanesini secin:
```

```
simpson 1/3 kurali (0):
```

```
simpson 3/8 kurali (1):
```

```
1
```

```
Fonksiyonu girin: 3*x^2 + 2*x
```

```
lutfen baslangic degerini giriniz: 1
```

```
lutfen bitis degerini giriniz: 3
```

```
simpson 3/8 kuralina gore intergral = 34,0000
```


Trapezoidal Rule

Fonksiyonu gireriz.
Başlangıç ve bitiş değerlerini gireriz.

```
TRAPEZOIDAL RULE ILE INTEGRAL BULMA  
Fonksiyonu girin:  $3*x^2 + 2*x$   
lutfen baslangic degerini giriniz: 1  
lutfen bitis degerini giriniz: 3  
  
Yamuk kuralina gore integral = 38,0000
```

Gregory Newton

X noktalarımızı ve Y noktalarımızı gireriz.
Bize Gregory Newton interpolasyon polinomunu gösterir.
Sonra hesaplanmasını istediğimiz değeri gireriz.

GREGORY NEWTON YONTEMI ILE DEGER BULMA

lutfen ilk noktanin x kordinatini girin :1

lutfen ilk noktanin y kordinatini girin :3

lutfen ikinci noktanin x kordinatini girin :2

lutfen ikinci noktanin y kordinatini girin :5

lutfen ucuncu noktanin x kordinatini girin :6

lutfen ucuncu noktanin y kordinatini girin :10

lutfen dorduncu noktanin x kordinatini girin :10

lutfen dorduncu noktanin y kordinatini girin :35

| x | = | 1,000000 | 2,000000 | 6,000000 | 10,000000 |

|F(x)| = | 3,000000 | 5,000000 | 10,000000 | 35,000000 |

$3,00 + 2,00 * (x - 1,00) + -0,15 * (x - 1,00) * (x - 2,00) + 0,09 * (x - 1,00) * (x - 2,00) * (x - 6,00)$

lutfen degerini hesaplamak istediginiz noktayi girin :4

degerimiz = 7,0667

