

YILDIZ TEKNİK ÜNİVERSİTESİ
ELEKTRİK-ELEKTRONİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



Sayısal Analiz Projesi

Burak Başol

24011037

Öğretim Görevlisi

Öğr. Gör. Dr. Ahmet ELBİR

İstanbul

2025

İçindekiler

Proje Açıklaması	4
Ana Menü	5
Desteklenen Fonksiyonlar.....	6
Desteklenen İfadeler	6
Polinom	6
Üstel.....	6
Logaritmik.....	6
Trigonometrik	7
Ters Trigonometrik	7
Örnek	7
Matris Girişi	8
Bisection Yöntemi	9
Girdiler	9
Örnek	9
Regula-Falsi Yöntemi.....	10
Girdiler	10
Örnek	10
Newton-Raphson Yöntemi	11
Girdiler	11
Örnek	11
Matrisin Tersi	12
Girdiler	12
Örnek	12
Cholesky (ALU) Yöntemi.....	13
Girdiler	13
Örnek	13
Gauss Seidal Yöntemi.....	14
Girdiler	14

Örnek	14
Sayısal Türev.....	15
Girdiler	15
Örnek	15
Simpson Yöntemi	16
Girdiler	16
Örnek	16
Trapez Yöntemi	17
Girdiler	17
Örnek	17
Gregory Newton Enterpolasyonu	17
Girdiler	17
Örnek	17

Proje Açıklaması

Proje dokümanında istenen şekilde girdi alınması sağlanmış, metin matematiksel formüllere dönüştürülmüş ve seçilen yöntemle göre formül uygulanarak uygun çözüm üretilmiştir.

Aşağıda yapılan metodlar sırası ile verilmiştir.

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

Yöntemlerin yapılıp yapılmadığını gösteren tablo:

[illegible]

Ana Menü

Ana menüde istenilen yöntemlerden birinin seçilmesi istenmektedir.

- 1. Bisection Method
 - 2. Regula-Falsi Method
 - 3. Newton-Raphson Method
 - 4. Inverse of an NxN Matrix
 - 5. Cholesky Method
 - 6. Gauss-Seidel Method
 - 7. Numerical Derivative
 - 8. Simpson's Method
 - 9. Trapezoidal Method
 - 10. Gregory-Newton Interpolation
- Write the index of the method u want to use:

Desteklenen Fonksiyonlar

Desteklenen İfadeler

- Sabit Sayılar: 7, 3
- Değişken: x
- Parantez: (,)
- Euler sayısı: e (Örnek: e^x)
- Pi sayısı: π (Örnek: π^x)
- Toplama: + (Örnek: $x + 5$)
- Çıkarma: - (Örnek: $x - 3$)
- Çarpma: * (Örnek: 2^x , $2x$)
- Bölme: / (Örnek: $x / 4$)
- Üs alma: ^ (Örnek: x^3)

Polinom

Polinomsal ifadeler aşağıdaki şekildi girilebilir.

- $x^3 + x^2 + 3x + 3.5$

Üstel

Üs alma operatörü kullanılarak değişkenlerin üssel ifadeleri ya da e üzeri ifadeler alınabilir.

- 3^x
- e^x
- $\exp(x)$

Logaritmik

Logaritmik ifadeler aşağıda belirtilen şekillerde girilebilir.

- $\log(x)$: \log ifadesi \ln ile aynı işlevi görmektedir.
- $\ln(x)$
- $\log_x(5+x)$
- $\log_5(x)$

Trigonometrik

Trigonometrik ifadeler kullanılabilir.

- $\sin(x)$
- $\cos(x)$
- $\tan(x)$
- $\cot(x)$
- $\sec(x)$
- $\csc(x)$

Ters Trigonometrik

Ters trigonometrik ifadeler desteklenmektedir.

- $\arcsin(x)$
- $\arccos(x)$
- $\arctan(x)$
- $\text{arccot}(x)$
- $\text{arcsec}(x)$
- $\text{arccsc}(x)$

Örnek

Aşağıdaki formüller tüm ifadeleri içermektedir. Ayrıca girdilerde boşluklar göz ardı edilmektedir. Herhangi bir yere boşluk koyabilirsiniz.

- $\log_x((\sin(\pi x^2 - e) + 5 \cos(xe / (2 + x)))^{1/3}) - \tan(\exp(0.5x) - \log(x+10)) + (2x + \pi)^{1.5}$
- $\text{asec}((x^3 - 2x + 5) / (x^2 + 1)) + \log_2((\sin(x\pi) - \cos(e^x))^2 + 1.1) - ((x+e)/(x-\pi))^{0.75} + \text{acot}(x)$
- $\csc((\log(x^2+e) + \exp(-x)) / (x - \pi + 1.5)) - ((\sin(x) + \cos(x))^2 + \log_x(e^{2.1})) / (\arctan(x/\pi) + 3.2)^{1/e}$

Matris Girişi

Matris üzerinde çalışacak olan yöntemlerden birisi seçildiği zaman matris girdisi alınmaktadır.

Matrise ait değerler yan yana girilebileceği gibi satır satır da girilebilir.

```
Write the size N of the NxN matrix:
5
Write the matrix's values (row by row, 25 values total for 5x5 matrix):
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
The matrix you provided (5x5):
0.000000 1.000000 2.000000 3.000000 4.000000
5.000000 6.000000 7.000000 8.000000 9.000000
10.000000 11.000000 12.000000 13.000000 14.000000
15.000000 16.000000 17.000000 18.000000 19.000000
20.000000 21.000000 22.000000 23.000000 24.000000
```


Bisection Yöntemi

Girdiler

- Formül
- Hata payı
- Başlangıç değeri
- Bitiş değeri

Örnek

Aşağıdaki örnekte $\exp(x) - x - 2$ formülü kullanılmıştır. Hata payı 10^{-5} , başlangıç değeri 1 ve bitiş değeri 2'dir.

Yaklaşık kök: 1.14619446

```
Write the equation: ( ^ is power operator, _ in log_ for base)
Supported constants: e, pi. Supported variable: x.
Functions: sin, cos, tan, cot, sec, csc, asin, acos, atan, acot, asec, acsc, exp, log (natural), ln (natural), log_BASE (e.g., log_10, log_x)
exp(x) - x - 2
Enter desired error tolerance exponent (e.g., 6 for 10^-6): 5
Using error tolerance: 1.00e-05
Enter interval endpoints a and b: 1 2
```

Iter	a	f(a)	b	f(b)	m	f(m)	b-a
0	1.00000e+00	-2.81718e-01	2.00000e+00	3.38906e+00	-	-	1.00e+00
1	1.00000e+00	-2.81718e-01	2.00000e+00	3.38906e+00	1.50000e+00	9.81689e-01	1.00e+00
2	1.00000e+00	-2.81718e-01	1.50000e+00	9.81689e-01	1.25000e+00	2.40343e-01	5.00e-01
3	1.00000e+00	-2.81718e-01	1.25000e+00	2.40343e-01	1.12500e+00	-4.47832e-02	2.50e-01
4	1.12500e+00	-4.47832e-02	1.25000e+00	2.40343e-01	1.18750e+00	9.13738e-02	1.25e-01
5	1.12500e+00	-4.47832e-02	1.18750e+00	9.13738e-02	1.15625e+00	2.17434e-02	6.25e-02
6	1.12500e+00	-4.47832e-02	1.15625e+00	2.17434e-02	1.14062e+00	-1.19018e-02	3.12e-02
7	1.14062e+00	-1.19018e-02	1.15625e+00	2.17434e-02	1.14844e+00	4.82459e-03	1.56e-02
8	1.14062e+00	-1.19018e-02	1.14844e+00	4.82459e-03	1.14453e+00	-3.56257e-03	7.81e-03
9	1.14453e+00	-3.56257e-03	1.14844e+00	4.82459e-03	1.14648e+00	6.25007e-04	3.91e-03
10	1.14453e+00	-3.56257e-03	1.14648e+00	6.25007e-04	1.14551e+00	-1.47028e-03	1.95e-03
11	1.14551e+00	-1.47028e-03	1.14648e+00	6.25007e-04	1.14600e+00	-4.23011e-04	9.77e-04
12	1.14600e+00	-4.23011e-04	1.14648e+00	6.25007e-04	1.14624e+00	1.00904e-04	4.88e-04
13	1.14600e+00	-4.23011e-04	1.14624e+00	1.00904e-04	1.14612e+00	-1.61077e-04	2.44e-04
14	1.14612e+00	-1.61077e-04	1.14624e+00	1.00904e-04	1.14618e+00	-3.00923e-05	1.22e-04
15	1.14618e+00	-3.00923e-05	1.14624e+00	1.00904e-04	1.14621e+00	3.54044e-05	6.10e-05
16	1.14618e+00	-3.00923e-05	1.14621e+00	3.54044e-05	1.14619e+00	2.65567e-06	3.05e-05

```
Approximate root: 1.14619446
f(root) = 2.66e-06, iterations = 16
If you want to use another method, type 1 (or any other number to exit):
```

Regula-Falsi Yöntemi

Girdiler

- Formül
- Hata payı
- Başlangıç değeri
- Bitiş değeri

Örnek

Aşağıdaki örnekte $x \cdot \log_{10}(x) - 1.2$ formülü kullanılmıştır. Hata payı 10 üzeri -5, başlangıç değeri 2 ve bitiş değeri 3'dür.

Yaklaşık kök: 2.74063626

```
Write the equation: ( ^ is power operator, _ in log_ for base)
Supported constants: e, pi. Supported variable: x.
Functions: sin, cos, tan, cot, sec, csc, asin, acos, atan, acot, asec, acsc, exp, log (natural), ln (natural), log_BASE (e.g., log_10, log_x)
x*log_10(x) - 1.2
Enter desired error tolerance exponent (e.g., 6 for 10^-6): 5
Using error tolerance: 1.00e-05
Enter interval endpoints a and b: 2 3

Iter | a | f(a) | b | f(b) | m | f(m) | |m-prev_m| | b-a
-----
0 | 2.00000e+00 | -5.97940e-01 | 3.00000e+00 | 2.31364e-01 | - | - | - | 1.00e+00
1 | 2.00000e+00 | -5.97940e-01 | 3.00000e+00 | 2.31364e-01 | 2.72101e+00 | -1.70911e-02 | nan | 1.00e+00
2 | 2.72101e+00 | -1.70911e-02 | 3.00000e+00 | 2.31364e-01 | 2.74021e+00 | -3.84056e-04 | 1.92e-02 | 2.79e-01
3 | 2.74021e+00 | -3.84056e-04 | 3.00000e+00 | 2.31364e-01 | 2.74064e+00 | -8.58134e-06 | 4.31e-04 | 2.60e-01
Convergence: |f(m)| < tolerance.
-----
Approximate root: 2.74063626
f(root) = -8.58e-06, iterations = 3
If you want to use another method, type 1 (or any other number to exit):
```

Newton-Raphson Yöntemi

Girdiler

- Formül
- Hata payı
- Başlangıç değeri

Örnek

Aşağıdaki örnekte $\cos(x)-x^2$ formülü kullanılmıştır. Hata payı 10^{-6} ve başlangıç değeri 1'dir.

Yaklaşık kök: 0.82413232

```
Write the equation: ( ^ is power operator, _ in log_ for base)
Supported constants: e, pi. Supported variable: x.
Functions: sin, cos, tan, cot, sec, csc, asin, acos, atan, acot, asec, acsc, exp, log (natural), ln (natural), log_BASE (e.g., log_10, log_x)
cos(x) - x^2
Enter desired error tolerance exponent (e.g., 6 for 10^-6): 6
Using error tolerance: 1.00e-06
Enter initial guess x0: 1

Iter |  x_curr  |  f(x_curr) | df(x_curr) |  x_next  | |x_next-x_curr|
-----
  1 | 1.00000e+00 | -4.59698e-01 | -2.84147e+00 | 8.38218e-01 | 1.62e-01
  2 | 8.38218e-01 | -3.38217e-02 | -2.41989e+00 | 8.24242e-01 | 1.40e-02
  3 | 8.24242e-01 | -2.61003e-04 | -2.38252e+00 | 8.24132e-01 | 1.10e-04
  4 | 8.24132e-01 | -1.60762e-08 | -2.38222e+00 | 8.24132e-01 | 6.75e-09
Convergence: |f(x_current)| < tolerance.

Approximate root: 0.82413232
f(root) = -1.61e-08, iterations = 4
If you want to use another method, type 1 (or any other number to exit):
```

Matrisin Tersi

Girdiler

- Boyut
- Matris elemanları

Örnek

Aşağıdaki örnekte 3x3'lük bir matris kullanılmıştır. Matrise ait değerler sırası ile 2 -1 0 -1 2 -1 0 -1 2 olarak verilmiştir.

Matrisin tersi: 0.75 0.50 0.25 0.50 1.00 0.50 0.25 0.50 0.75

```
Write the size N of the NxN matrix:
3
Write the matrix's values (row by row, 9 values total for 3x3 matrix):
2 -1 0 -1 2 -1 0 -1 2
Initial Augmented Matrix [A|I]:
  2.0000  -1.0000  0.0000  1.0000  0.0000  0.0000
 -1.0000  2.0000 -1.0000  0.0000  1.0000  0.0000
  0.0000  -1.0000  2.0000  0.0000  0.0000  1.0000
-----
Normalizing pivot row 0 (dividing by 2.0000)
Eliminating in row 1 using row 0 (factor -1.0000)
Eliminating in row 2 using row 0 (factor 0.0000)
Matrix after processing column 0:
  1.0000  -0.5000  0.0000  0.5000  0.0000  0.0000
  0.0000  1.5000 -1.0000  0.5000  1.0000  0.0000
  0.0000  -1.0000  2.0000  0.0000  0.0000  1.0000
-----
Normalizing pivot row 1 (dividing by 1.5000)
Eliminating in row 0 using row 1 (factor -0.5000)
Eliminating in row 2 using row 1 (factor -1.0000)
Matrix after processing column 1:
  1.0000  0.0000 -0.3333  0.6667  0.3333  0.0000
  0.0000  1.0000 -0.6667  0.3333  0.6667  0.0000
  0.0000  0.0000  1.3333  0.3333  0.6667  1.0000
-----
Normalizing pivot row 2 (dividing by 1.3333)
Eliminating in row 0 using row 2 (factor -0.3333)
Eliminating in row 1 using row 2 (factor -0.6667)
Matrix after processing column 2:
  1.0000  0.0000  0.0000  0.7500  0.5000  0.2500
  0.0000  1.0000  0.0000  0.5000  1.0000  0.5000
  0.0000  0.0000  1.0000  0.2500  0.5000  0.7500
-----
Final Inverted Matrix (A^-1 from right side of [I|A^-1]):
  0.750000  0.500000  0.250000
  0.500000  1.000000  0.500000
  0.250000  0.500000  0.750000
If you want to use another method, type 1 (or any other number to exit):
```

Cholesky (ALU) Yöntemi

Girdiler

- Boyut
- Matris elemanları

Örnek

Aşağıdaki örnekte 3x3'lük bir matris kullanılmıştır. Matrise ait değerler sırası ile 4 12 -16 12 37 -43 -16 -43 98 olarak verilmiştir.

L matrisi: 2.00 0.00 0.00 6.00 1.00 0.00 -8.00 5.00 3.00

```
Write the size N of the NxN matrix:
3
Write the matrix's values (row by row, 9 values total for 3x3 matrix):
4 12 -16 12 37 -43 -16 -43 98
Original Matrix A (will be transformed into L):
  4.0000    12.0000   -16.0000
 12.0000    37.0000  -43.0000
-16.0000   -43.0000   98.0000
-----
Matrix L after processing column 0:
  2.0000    0.0000    0.0000
  6.0000    37.0000    0.0000
 -8.0000   -43.0000   98.0000
-----
Matrix L after processing column 1:
  2.0000    0.0000    0.0000
  6.0000    1.0000    0.0000
 -8.0000    5.0000   98.0000
-----
Matrix L after processing column 2:
  2.0000    0.0000    0.0000
  6.0000    1.0000    0.0000
 -8.0000    5.0000    3.0000
-----
Final L Matrix (Lower Triangular from Cholesky  $A = L \cdot L^T$ ):
  2.000000    0.000000    0.000000
  6.000000    1.000000    0.000000
 -8.000000    5.000000    3.000000
If you want to use another method, type 1 (or any other number to exit):
```

Gauss Seidal Yöntemi

Girdiler

- Matris boyutu
- Matris elemanları
- Hata payı

Örnek

Aşağıdaki örnekte 3x3 matris kullanılmıştır. Hata payı olarak 10 üzeri -5 girilmiştir. Matrise ait değerler girilmiştir.

Sonuç:

- x0: 3.876401
- x1: 5.235953
- x2: -3.269661

```
Enter the size N (number of rows/equations) for the augmented matrix [A|b]:
3
Enter the augmented matrix [A|b] values, row by row (3 rows, 4 columns):
4 -1 1 7 -1 6 2 21 1 2 5 -2
Enter desired error tolerance exponent for Gauss-Seidel (e.g., 6 for 10^-6): 5
Using error tolerance: 1.00e-05

Iter |   x[0]   |   x[1]   |   x[2]   |   max_diff
-----|-----|-----|-----|-----
  1 | 1.75000e+00 | 3.79167e+00 | -2.26667e+00 | 3.79e+00
  2 | 3.26458e+00 | 4.79965e+00 | -2.97278e+00 | 1.51e+00
  3 | 3.69311e+00 | 5.10644e+00 | -3.18120e+00 | 4.29e-01
  4 | 3.82191e+00 | 5.19738e+00 | -3.24334e+00 | 1.29e-01
  5 | 3.86018e+00 | 5.22448e+00 | -3.26183e+00 | 3.83e-02
  6 | 3.87158e+00 | 5.23254e+00 | -3.26733e+00 | 1.14e-02
  7 | 3.87497e+00 | 5.23494e+00 | -3.26897e+00 | 3.39e-03
  8 | 3.87598e+00 | 5.23565e+00 | -3.26946e+00 | 1.01e-03
  9 | 3.87628e+00 | 5.23586e+00 | -3.26960e+00 | 3.01e-04
 10 | 3.87637e+00 | 5.23593e+00 | -3.26964e+00 | 8.94e-05
 11 | 3.87639e+00 | 5.23595e+00 | -3.26966e+00 | 2.66e-05
 12 | 3.87640e+00 | 5.23595e+00 | -3.26966e+00 | 7.93e-06

Gauss-Seidel converged in 12 iterations.

Final Solution x:
x0 = 3.876401
x1 = 5.235953
x2 = -3.269661
If you want to use another method, type 1 (or any other number to exit):
```

Sayısal Türev

Girdiler

- Formül
- Türevi alınacak kök
- h değeri

Örnek

Aşağıdaki örnekte $\exp(x) \cdot \sin(x)$ fonksiyonu girilmiştir. Hesaplanacak değer olarak 1 ve h değeri olarak 0.001 girilmiştir.

İleri türev: 3.75751765

Merkezi türev: 3.75604895

Geri türev: 3.75458026

```
Write the equation: ( ^ is power operator, _ in log_ for base)
Supported constants: e, pi. Supported variable: x.
Functions: sin, cos, tan, cot, sec, csc, asin, acos, atan, acot, asec, acsc, exp, log (natural), ln (natural), log_BASE (e.g., log_10, log_x)
exp(x)*sin(x)
Enter point x0 and step size h (e.g., 1.0 0.001): 1 0.001
Forward difference derivative at x0=1.0000 (h=0.001): 3.75751765
Backward difference derivative at x0=1.0000 (h=0.001): 3.75458026
Central difference derivative at x0=1.0000 (h=0.001): 3.75604895
If you want to use another method, type 1 (or any other number to exit):
```

Simpson Yöntemi

Girdiler

- Formül
- Başlangıç noktası
- Bitiş noktası
- 1/3 yöntemi için n değeri
- 3/8 yöntemi için n değeri

Örnek

Aşağıdaki örnekte $1 / (1 + x^2)$ formülü girilmiştir. Başlangıç noktası olarak 0, bitiş noktası olarak 1, 1/3 kuralı için n değeri olarak 10 ve 3/8 kuralı için n değeri olarak 9 girilmiştir.

Simpson 1/3 kuralı sonucu: 0.78539815

Simpson 3/8 kuralı sonucu: 0.78539808

```
Write the equation: ( ^ is power operator, _ in log_ for base)
Supported constants: e, pi. Supported variable: x.
Functions: sin, cos, tan, cot, sec, csc, asin, acos, atan, acot, asec, acsc, exp, log (natural), ln (natural), log_BASE (e.g., log_10, log_x)
1 / (1 + x^2)
Enter interval endpoints a and b for integration: 0 1
--- Simpson's 1/3 Rule ---
Enter number of subintervals n for 1/3 rule (must be EVEN and >= 2): 10
Simpson's 1/3 rule result (n=10): 0.78539815
--- Simpson's 3/8 Rule ---
Enter number of subintervals n for 3/8 rule (must be a MULTIPLE OF 3 and >=3): 9
Simpson's 3/8 rule result (n=9): 0.78539808
If you want to use another method, type 1 (or any other number to exit):
```


Trapez Yöntemi

Girdiler

- Formül
- Başlangıç noktası
- Bitiş noktası
- Alt aralık sayısı

Örnek

Aşağıdaki örnekte $\exp(-x^2)$ formülü girilmiştir. Başlangıç noktası olarak 0, bitiş noktası olarak 1 ve aralık olarak 20 girilmiştir.

Sonuç: 0.74667084

```
Write the equation: ( ^ is power operator, _ in log_ for base)
Supported constants: e, pi. Supported variable: x.
Functions: sin, cos, tan, cot, sec, csc, asin, acos, atan, acot, asec, acsc, exp, log (natural), ln (natural), log_BASE (e.g., log_10, log_x)
exp(-x^2)
Enter interval endpoints a and b, and number of subintervals n (n >= 1): 0 1 20
Trapezoidal rule result (n=20): 0.74667084
If you want to use another method, type 1 (or any other number to exit):
```

Gregory Newton Enterpolasyonu

Girdiler

- Nokta sayısı
- Noktaların x ve y değerleri
- Enterpolasyon noktası

Örnek

Aşağıdaki örnekte 4 noktaya ait değerler girilmiştir ve 1.5 sayısının enterpolasyon değerinin hesaplanması istenmiştir.

Sonuç: 1.375

```
Enter number of data points n (>= 2): 4
Enter x[i] and y[i] pairs (assuming x values are equally spaced):
Point 0 (x y): 0 1
Point 1 (x y): 1 0
Point 2 (x y): 2 5
Point 3 (x y): 3 22
Enter interpolation point X: 1.5
Interpolated value P(1.5000) = 1.375000
If you want to use another method, type 1 (or any other number to exit):
```