



Yıldız Teknik Üniversitesi
Elektrik-Elektronik Fakültesi
Bilgisayar Mühendisliği Bölümü

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

İsim:

No:

E-posta: _____

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 Entropolasyonu	20
Parametreler	20
Örnekler	20

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. $N \times N$ '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 enterpolasyonu

YÖNTEMLERİN YAPILIP YAPILMADIĞINI AŞAĞIDAKİ TABLODA GÖSTERİLDİĞİ GİBİ 1/0 OLARAK GÖSTERİNİZ									
1	2	3	4	5	6	7	8	9	10
1	1	1	1	1	1	1	1	1	1

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 enterpolasyon yöntemleri (10) için ilk istenilen parametre fonksiyondur. Bu fonksiyon sırasıyla polinom, üstel, logaritmik, trigonometrik ve ters trigonometrik fonksiyon tiplerini barındıracak şekilde ayarlanabilir. Her fonksiyon tipi için, o tipten kaç tane ifade bulunduğu girildikten sonra, girilen sayı kadar o tipin parametreleri girilir. 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

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 =
$$\begin{cases} f(x) \leq \epsilon, & 1 \\ \frac{\text{end} - \text{start}}{2^n} \leq \epsilon, & 2 \end{cases}$$

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 = $\frac{\text{end} - \text{start}}{2^n} \leq \epsilon$)

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
```

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
```