

SAYISAL ÇÖZÜM YÖNTEMLERİ DERSİ

7.ÖDEV

Alırıza BİLİR 18014125

Kodlar:

```
////GausQuadn.m////
```

```
function [I] = GausQuadn(f,a,b,n)
```

```
%f=inline (f);
```

```
%n;
```

```
I=0;
```

```
if n == 2
```

```
    w(1) = 1; x(1) = 0.5773502691896257;
```

```
    w(2) = 1; x(2) = -0.5773502691896257;
```

```
    for i=1:n
```

```
        I=I+(b-a)*0.5*w(i)*f((b-a)*0.5*x(i)+(b+a)*0.5);
```

```
    end
```

```
elseif n == 3
```

```
    w(1) = 0.5555555555555556; x(1) = 0.7745966692414834;
```

```
    w(2) = 0.6521451548625460; x(2) = 0;
```

```
    w(3) = 0.5555555555555556; x(3) = -0.77459666924148347;
```

```
    for i=1:n
```

```
        I=I+(b-a)*0.5*w(i)*f((b-a)*0.5*x(i)+(b+a)*0.5);
```

```
    end
```

```
elseif n == 4
```

```
    w(1) = 0.3478548451374544; x(1) = 0.8611363115940525;
```

```
    w(2) = 0.6521451548625460; x(2) = 0.3399810435848563;
```

```
    w(3) = 0.6521451548625460; x(3) = -0.3399810435848563;
```

```
    w(4) = 0.3478548451374544; x(4) = -0.8611363115940525;
```

```
    for i=1:n
```

```
        I=I+(b-a)*0.5*w(i)*f((b-a)*0.5*x(i)+(b+a)*0.5);
```

```
    end
```

```
elseif n == 5
```

```
    w(1) = 0.5688888888888889; x(1) = 0;
```

```
    w(2) = 0.4786286704993665; x(2) = 0.5384693101056831;
```

```
    w(3) = 0.4786286704993665; x(3) = -0.5384693101056831;
```

```
    w(4) = 0.2369268850561891; x(4) = 0.906179845938664;
```

```
    w(5) = 0.2369268850561891; x(5) = -0.906179845938664;
```

```
    for i=1:n
```

```
        I=I+(b-a)*0.5*w(i)*f((b-a)*0.5*x(i)+(b+a)*0.5);
```

```
    end
```

```
elseif n == 6
```

```
    w(1) = 0.3607615730481386; x(1) = 0.6612093864662645;
```

```
    w(2) = 0.3607615730481386; x(2) = -0.6612093864662645;
```

```
    w(3) = 0.4679139345726910; x(3) = -0.2386191860831969;
```

```
    w(4) = 0.4679139345726910; x(4) = 0.2386191860831969;
```

```
    w(5) = 0.1713244923791704; x(5) = -0.9324695142031521;
```

```
    w(6) = 0.1713244923791704; x(6) = 0.9324695142031521;
```

```

        for i=1:n
            I=I+(b-a)*0.5*w(i)*f((b-a)*0.5*x(i)+(b+a)*0.5);
        end

elseif n == 7
    w(1) = 0.4179591836734694; x(1) = 0;
    w(2) = 0.3818300505051189; x(2) = 0.4058451513773972;
    w(3) = 0.3818300505051189; x(3) = -0.4058451513773972;
    w(4) = 0.2797053914892766; x(4) = -0.7415311855993945;
    w(5) = 0.2797053914892766; x(5) = 0.7415311855993945;
    w(6) = 0.1294849661688697; x(6) = -0.9491079123427585;
    w(7) = 0.1294849661688697; x(7) = 0.9491079123427585;
    for i=1:n
        I=I+(b-a)*0.5*w(i)*f((b-a)*0.5*x(i)+(b+a)*0.5);
    end

elseif n == 8
    w(1) = 0.3626837833783620; x(1) = -0.1834346424956498;
    w(2) = 0.3626837833783620; x(2) = 0.1834346424956498;
    w(3) = 0.3137066458778873; x(3) = -0.5255324099163290;
    w(4) = 0.3137066458778873; x(4) = 0.5255324099163290;
    w(5) = 0.2223810344533745; x(5) = -0.7966664774136267;
    w(6) = 0.2223810344533745; x(6) = 0.7966664774136267;
    w(7) = 0.1012285362903763; x(7) = -0.9602898564975363;
    w(8) = 0.1012285362903763; x(8) = 0.9602898564975363;
    for i=1:n
        I=I+(b-a)*0.5*w(i)*f((b-a)*0.5*x(i)+(b+a)*0.5);
    end

else
    disp('This version work only up to 8 points Gaussian quadrature')
end

return ;
end

////testIntegrals.m////

function testIntegrals()
    f = @(x) sin(x); % Örnek fonksiyon (kendi fonksiyonunuzu buraya tanımlayabilirsiniz)
    a = 8;
    b = -12;
    real_integral = integral(f, a, b); % Gerçek integral değeri (a ve b sınırlarını belirtin)
    tolerance = 1e-6; % Hata toleransı

    max_error = tolerance + 1;
    prev_error = max_error;
    count = 0;

    errors = zeros(1, 7); % Hataları saklamak için bir dizi oluşturuluyor
    integrals = zeros(1, 7); % Integral değerlerini saklamak için bir dizi oluşturuluyor

    for i = 2:8 % 2 noktalıdan 8 noktaliya kadar hesaplamalar yapılıyor
        I_gaussian = GausQuadn(f, a, b, i);
        integral_value = I_gaussian; % Quadn ile hesaplanan integral değeri

        error = abs(real_integral - integral_value);
        errors(i-1) = error; % Hataları diziye atıyoruz
        integrals(i-1) = integral_value; % Integral değerlerini diziye atıyoruz

        disp(['Gaussian Quadrature (', num2str(i), ' Nokta) Hatası: ', num2str(error)]);
    end
end

```

```

disp(['Gaussian Quadrature (', num2str(i), ' Nokta) İntegral Değeri: ',
num2str(integral_value)]);

if error >= prev_error || error == prev_error
    count = count + 1;
else
    count = 0; % Eğer hata azaldıysa sayacı sıfırla
end

if count > 5
    disp('Hata değeri art arda arttı veya aynı, döngü sonlandırılıyor.');
```

```

    break;
end

prev_error = error; % Önceki hatayı güncelle
end

disp('Gerçek İntegral Değeri:');
disp(real_integral); % Gerçek integrali ekrana yazdır
disp('Tüm Hatalar:');
disp(errors); % Hataları ekrana yazdır
disp('Tüm İntegral Değerleri:');
disp(integrals); % Hesaplanan integral değerlerini ekrana yazdır
end

```

Sonuçlar:

```

>> testIntegrals
Gaussian Quadrature (2 Nokta) Hatası: 16.8639
Gaussian Quadrature (2 Nokta) İntegral Değeri: 15.8745
Gaussian Quadrature (3 Nokta) Hatası: 10.1612
Gaussian Quadrature (3 Nokta) İntegral Değeri: 9.1718
Gaussian Quadrature (4 Nokta) Hatası: 14.8235
Gaussian Quadrature (4 Nokta) İntegral Değeri: -15.8128
Gaussian Quadrature (5 Nokta) Hatası: 7.5552
Gaussian Quadrature (5 Nokta) İntegral Değeri: 6.5659
Gaussian Quadrature (6 Nokta) Hatası: 2.0965
Gaussian Quadrature (6 Nokta) İntegral Değeri: -3.0859
Gaussian Quadrature (7 Nokta) Hatası: 0.37661
Gaussian Quadrature (7 Nokta) İntegral Değeri: -0.61275
Gaussian Quadrature (8 Nokta) Hatası: 0.047941
Gaussian Quadrature (8 Nokta) İntegral Değeri: -1.0373
Gerçek İntegral Değeri:
-0.9894

Tüm Hatalar:
    16.8639    10.1612    14.8235     7.5552     2.0965     0.3766     0.0479

Tüm İntegral Değerleri:
    15.8745     9.1718   -15.8128     6.5659    -3.0859    -0.6127    -1.0373

```