

Mahmut alomeyr

19011920

Yapılan yöntemler:

1. Bisection
2. Regula-Falsi
3. Newton-Rapshon
4. $N \times N$ 'lik bir matrisin tersi
5. Gauss Eleminasyon
6. Gauss Seidal
7. Sayısal Türev (merkezi, ileri ve geri)
8. Simpson yöntemi
9. Trapez yöntemi
10. Değişken dönüşümsüz Gregory Newton Enterpolasyonu

Note : bütün yöntemler yapıldı ama bu dosyaya sadece sunumda gösterdiğim yöntemleri koydum.

1.Bisection

```
#include<stdio.h>
#include<math.h>

float f(float[],int , float);

int main()
{
    int i,j;
    float polinom[100];
    int n,m=2;
    float x1,x2,x3,x5;
    float hata;
    printf("polinomun en buyuk derecesini giriniz : ");
    scanf("%d",&n);
    for(i=0 ; i<=n ; i++)
    {
        printf("%d inci terimin katsayisini giriniz : ",i);
        scanf("%f",&polinom[i]);
    }
    printf("aralik giriniz ");
    printf("\n soldan : ");
    scanf("%f",&x1);

    printf("sagdan :");
    scanf("%f",&x2);

    printf("hata degerini giriniz : ");
    scanf("%f",&hata);
    i=1;
    x3=(x1+x2)/2;
    if(f(polinom , n , x1)*f(polinom,n,x2)<0){
        x5=(x1-x2)/m;
        if( fabs(x5) > hata){
            if(f(polinom , n , x3) != 0){
```

```

        do{
            printf("\n %d iterasyon _____kok:%0.10f",i,(x1+x2)/2);
            x3=(x1+x2)/2;
            m=m*2;
            if(f(polinom , n , x3)*f(polinom,n,x2)<0){
                x1=x3;
            }else{
                x2=x3;
            }

            i++;
            x5=(x1-x2)/m;
        }while(fabs(x5)>hata);

    }else{
        printf("%i iterasyon _____ kok : %0.10f",i,x3);
    }

}

}

printf(" \n son kok : %0.10f",(x1+x2)/2);

}else{
    printf("bu aralikta kok yok");
}

return 0;
}

float f(float polinom[] , int n , float x)
{
    int i,j;
    float x1;
    float toplam=0;

    toplam =polinom[0];
    for(i=1 ; i<=n ; i++)
    {
        x1=x;
        for(j=1 ; j<i ; j++)

```

```
    {  
        x1=x1*x;  
    }  
    toplam = toplam + x1*polinom[i];  
}  
return toplam;  
}
```

```
C:\Users\MONSTER\Desktop\proje>a.exe  
polinomun en buyuk derecesini giriniz : 3  
0 inci terimin katsayisini giriniz : -6  
1 inci terimin katsayisini giriniz : 14  
2 inci terimin katsayisini giriniz : -7  
3 inci terimin katsayisini giriniz : 1  
aralik giriniz  
soldan : 0  
sagdan :1  
hata degerini giriniz : 0.01  
  
1 iterasyon _____kok:0.5000000000  
2 iterasyon _____kok:0.7500000000  
3 iterasyon _____kok:0.6250000000  
son kok : 0.5625000000  
C:\Users\MONSTER\Desktop\proje>
```

2. Değişken dönüşümsüz Gregory Newton Enterpolasyonu

```
#include<stdio.h>
int fakt(int );
int power(int , int );
float ilerifark( float [], int );
int isaret(int );
int kombinasyon(int , int );
float carpim(float [] , int , float );

int main()
{
    float X[100];
    float Y[100];
    float x0 ;
    int i,h;
    int n;
    float toplam,x;

    printf("x0 degerini giriniz = ");
    scanf("%f",&x0);

    printf("h degerini giriniz = ");
    scanf("%d",&h);

    printf("xlerini sayisini giriniz = ");
    scanf("%d",&n);

    for(i=0 ; i<n ; i++){
        X[i]=x0+(i*h);
    }

    for(i=0 ; i<n ; i++){
        printf("y%d degerini giriniz = ",i);
        scanf("%f",&Y[i]);
    }
    printf("x degerini giriniz = ");
    scanf("%f",&x);

    i=1;
```

```

toplam=Y[0];
while( ilerifark(Y,i)!= 0.0 && i<n){

    toplam += (ilerifark(Y,i) * carpim(X,i,x) )/(float)( fakt(i) * power(h,i)
) ;
    i++;

}
printf("\n %f",toplam);

}

int fakt(int i )
{
    int toplam=1;
    int j;
    for(j=1 ; j<=i ; j++)
    {
        toplam *= j;
    }

    return toplam;
}

int power(int h , int i)
{
    int toplam=1;
    int j;
    for(j=1 ; j<=i ;j++)
    {
        toplam *= h;
    }
    return toplam;
}

float ilerifark( float Y[], int i)
{
    float toplam=0;
    int j;
    for(j=0 ; j<=i ; j++)
    {
        toplam += isaret(j)*( kombinasyon(i,j) )* (Y[i-j]);
    }
}

```

```
    return toplam;
}

int isaret(int j)
{
    int toplam;
    if( j%2==0 ){
        return toplam=1;
    }
    return toplam = -1;
}

int kombinasyon(int i , int j )
{
    int toplam=0;
    toplam +=( fakt(i) )/( fakt(j)*fakt(i-j) );
    return toplam;
}

float carpim(float X[] , int i , float x)
{
    float toplam=1;
    int j;
    for(j=0 ; j<i ; j++)
    {
        toplam *= (x-X[j]);
    }
    return toplam;
}
```



```
C:\Users\MONSTER\Desktop\proje>a.exe
```

```
x0 degerini giriniz = 2
```

```
h degerini giriniz = 2
```

```
xlerini sayisini giriniz = 5
```

```
y0 degerini giriniz = 10
```

```
y1 degerini giriniz = 50
```

```
y2 degerini giriniz = 122
```

```
y3 degerini giriniz = 226
```

```
y4 degerini giriniz = 362
```

```
x degerini giriniz = 8
```

```
6.000000
```

```
24.000000
```

```
226.000000
```

```
C:\Users\MONSTER\Desktop\proje>
```