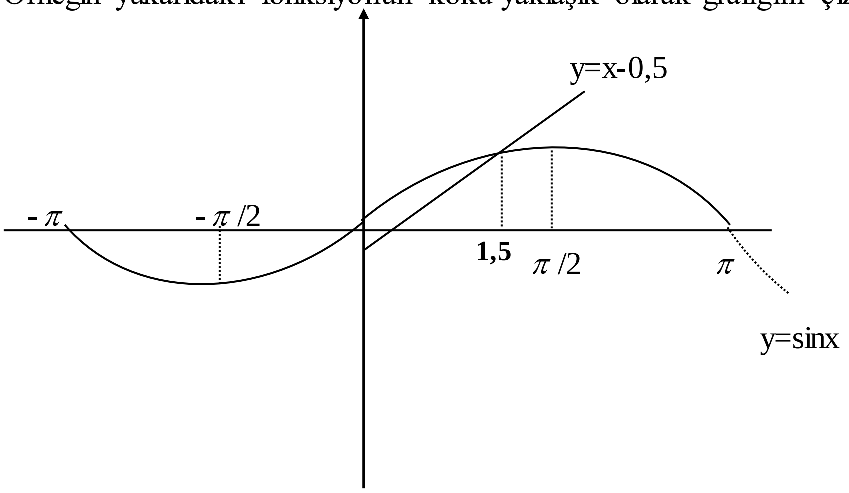
xx. BAZI SAYISAL YÖNTEMLER VE PROGRAMLARI

Derecesi 2'den fazla olan polinomlar ve üstel, logaritmik veya trigonometrik terimler içeren denklemlerin çözümünde sayısal yöntemler kullanılır. Örneğin

f (x) =sinx –x+ 0,5 =0

Çoğu durumda bu tür fonksiyonların analitik çözümleri yoktur veya varsa da analitik çözümlerin bulunması külfetlidir. Böyle durumlarda bilgisayar yardımıyla sayısal yöntemler kullanarak çözüme ulaşmak uygundur.

Örneğin yukarıdaki fonksiyonun kökü yaklaşık olarak grafiğini çizerek bulunabilir.

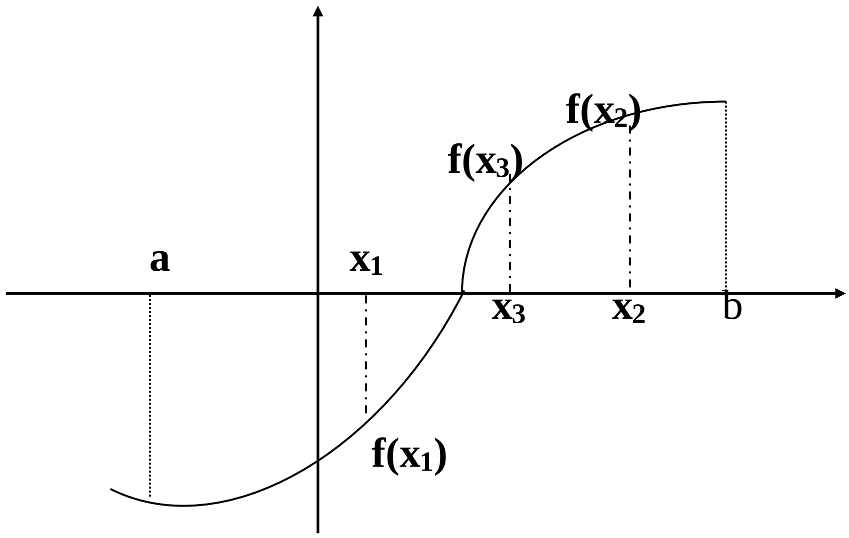


Kök yaklaşık olarak 1,5 civarındadır. Bir diğer imkan köke yaklaşan tahmini değerler vererek deneme yanılma ile yaklaşık kök bulunmasıdır:

x	1,5	1,45	1,49
f(x)	-0,0025	0,0427	0,0067

Buna göre aranan kök 1,49<x<1,50 aralığındadır ve kök x = 1,49 ± 0,005 şeklinde verilebilir. Kök x 0,01 hata ile bulunmuştur. Kökün istenen duyarlılıkta hesaplanması için çok sayıda sayısal yöntem geliştirilmiştir.

İkiye Bölme Yöntemi



Bir f(x) fonksiyonu için a<x<b aralığında f(a) < 0 ve f (b) > 0 olsun. Eğer

$$x = \frac{a+b}{2}$$

ise bu yeni x için (a, b aralığının ortasındaki değer) 3 ihtimal vardır:

1: $f\,(\frac{a+b}{2})$ = 0 ise bu durumda kök (a +b)/2 dir.

2: $f\,(\frac{a+b}{2}) < 0$ ise bu durumda kök $\frac{a+b}{2}$ ile b arasındadır

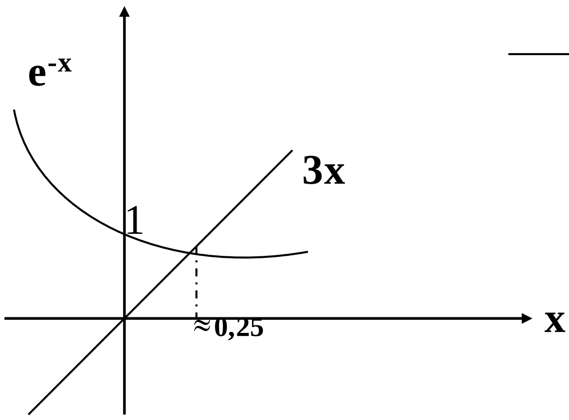
3: $f = (\frac{a+b}{2}) > 0$ ise bu durumda kök a ile $\frac{a+b}{2}$ arasındadır.

İstenen bir ε duyarlılığının erişilmesi için gereken adım sayısı:

$$n = \frac{b-a}{2^n} \leq \varepsilon$$

$$n = \frac{\ln(\frac{b-a}{\varepsilon})}{\ln 2}$$

Örnek: $f(x) = 3x - e^{-x} = 0$



x	0,25	0,27
f(x)	**-0,0028**	**0,0466**

$\in$ **= 0,001**

a	b	$x = \dfrac{a+b}{2}$	f(x)
0,25	0,27	0,26	0,0089
0,25	0,26	0,255	-0,0099
0,255	0,26	0,2575	-0,0005
0,2575	0,26	0,2588	0,0044
0,2575	0,2588	0,2582	0,0022

$$n = \frac{\ln \dfrac{(0,27 - 0,25)}{0,001}}{\ln 2} = 4,32$$

n = 5

$|0,2575 - 0,2581| = 0$ $0,0007 < \in = 0,001$

```
// İkiye bölme yöntemi
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
int main ()
{
float a,b,EPS,n,x,fx;
int I;
printf("a,b ve EPS degerlerini giriniz:\n");
scanf("%f%f%f",&a,&b,&EPS);
n=log((b-a)/EPS)/log(2);
ceil (n);
for (I=0;I<=n;I++)
{
	x=(a+b)/2;
	fx=(3*x)-exp(-x);
	if(fx>0)
		b=x;
	else if (fx<0)
		a=x;
	else
	{
		printf("x=%f kök bulundu.",fx);
		break ;
	}
printf("a=%f b=%f x=%f f(x)=%f\n",a,b,x,fx);
}
return 0;
}
```

Basit İterasyon Yöntemi

Bu yöntemde $f(x) = 0$ denklemi uygun şekilde bir $x = h(x)$ eşitliğine dönüştürülür ve iterasyon ile bu eşitliği sağlayan x istenen duyarlılıkta hesaplanır.

x_0 tahmini bir kök ise;
$x_1 = h(x_0)$
$x_2 = h(x_1)$
…
$x_n = h(x_{n-1})$

şeklinde ardışık hesaplamalarla (iterasyonla) her bir adımda yeni bir yaklaşık kök bulunur. İterasyon $|x_n - x_{n-1}| \leq \varepsilon$ olana kadar işlem sürdürülür. Burada ε istenen duyarlıktır ($\varepsilon \ll 1$).

Örnek:
$f(x) = 3x - e^{-x} = 0$ denkleminin kökünü $\varepsilon = 0,001$ duyarlıkla hesaplayınız.
Verilen fonksiyon f(x)

$x = e^{-x}/3 = h(x)$

şeklinde yazılabilir. Tahmini bir kök $x_0=0,25$ olsun. Buna göre
$x_1 = e^{-0,25}/3 = 0,2596$
$x_2 = e^{-0,2596}/3=0,2571$
$x_3 = e^{-0,2571}/3 = 0,2579$

İşlemler aşağıdaki tabloda sunulmuştur.

i	x_i	$x_{i+1}=h(x_i)$	$\lvert x_{i+1}-x_i \rvert$
0	0,25	0,2596	0,0096 > ε
1	0,2596	0,2571	0,0025> ε
2	0,2571	0,2578	0,0007< ε

Aranan kök = 0,2578 dir. n = 3adımda hesaplanmıştır.

Basit iterasyon yöntemi ile yakınsama için gereken şart h(x) fonksiyonunun türevinin mutlak değerinin 1'den küçük olmasıdır:

$|h'(x)| < 1$

Örnek: $f(x) = x^2 - x - 2 = 0$ denkleminin kökünü basit iterasyon yöntemi ile bulunuz.

$x = x^2 - 2 = h(x)$
$h'(x) = 2x$
$|h'(x)| >= 1$ ($|x| < 0.5$ hariç tüm diğer x değerleri için)

x	h(x)	$\lvert x_{i+1}-x_i \rvert$
3	7	$\varepsilon > 1$
7	47	$\varepsilon >> 1$
47	2207	

Hesaplamalar yakınsamamaktadır!

Eğer şöyle seçilirse x = $\sqrt{x+2}$

-2 < x < ∞ için

$$h^{\prime}(x) = \frac{1}{2\sqrt{x+2}} < 1$$

olur. Yani gerçek köklerin olduğu her x için türev 1'den küçüktür. Yakınsama için gereken şart yerine gelmiştir. Bu durumda işlemler aşağıdaki tabloda sunulmuştur.

x	h(x) = $\sqrt{x+2}$
3	2,236
2,336	2,058
2,058	2,014
2,014	2,003
…	…
	≈ 2

İşlem yakınsamakta olup aranan kök = 2 dir.

```
//Basit iterasyon
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
int main ()
{
float x,hx,EPS,I;
int K;
printf("f(x)=3x-e-x=0 denkleminin cozumu\n");
printf("Baslangic icin x degeri ve epsilon duyarliligi girin:");
scanf("%f%f",&x,&EPS);
hx=exp(-x)/3;
I=fabs(hx-x);
K=0;
printf("i Xi h(Xi) |Xi-Xi+1|\n");
printf("---------------------------------------------------------------------\n");
printf("%d %f %f %f\n",K,x,hx,I);
while(I>EPS)
{
        x=hx;
        hx=exp(-x)/3;
        I=fabs(x-hx);
        K++;
        printf("%d %f %f %f\n",K,x,hx,I);
}
printf("Aranan kok X=%f 'dir",hx);
return 0;
}
```