LAPORAN PEMROGRAMAN AIDED

PENYELESAIAN PERSAMAAN NON-LINEAR



Disusun Oleh:

Achmad Robith Fuadi 4210161027

PROGRAM STUDI TEKNOLOGI GAME

DEPARTEMEN TEKNOLOGI MULTIMEDIA KREATIF

POLITEKNIK ELEKTRONIKA NEGERI SURABAYA

2017

**Praktikum I Metode Tabel**

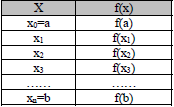
1. DASAR TEORI

Penyelesaian persamaan non-linear adalah penentuan akar-akar persamaan non linear dimana akar sebuah persamaan f(x) = 0 adalah nilai-nilai x yang menyebabkan nilai f(x) sama dengan nol. Persamaan f(x) adalah titik potong antara kurva f(x) dan sumbu x.

Theorema 1.1.

Suatu range *x=[a,b]* mempunyai akar bila *f(a) dan f(b)* berlawanan tanda atau memenuhi *f(a).f(b)<0*

Secara sederhana, untuk menyelesaikan persamaan non linier dapat dilakukan dengan menggunakan metode table atau pembagian area.Dimana untuk x = [*a*,*b*] atau x di antara *a* dan *b* dibagi sebanyak N bagian dan pada masing-masing bagian dihitung nilai *f(x)* sehingga diperoleh tabel :



Dari tabel ini, bila ditemukan *f(xk)=0* atau mendekati nol maka dikatakan bahwa *xk* adalah penyelesaian persamaan *f(xk)=0*.Bila tidak ada *f(xk)* yang sama dengan nol, maka dicari nilai *f(xk)* dan *f(xk+1)* yang berlawanan tanda, bila tidak ditemukan maka dikatakan tidak mempunyai akar untuk x = [*a*,*b*], dan bila ditemukan maka ada 2 pendapat untuk menentukan akar persamaan, yaitu :

1. Akar persamaan ditentukan oleh nilai mana yang lebih dekat, bila *|f(xk)|* ≤ *|f(xk+1)|* maka akarnya *xk*, dan bila *|f(xk+1)|<|f(xk)|* maka akarnya xk+1.

2. Akarnya perlu di cari lagi, dengan range x = [*xk* , *xk*+1]

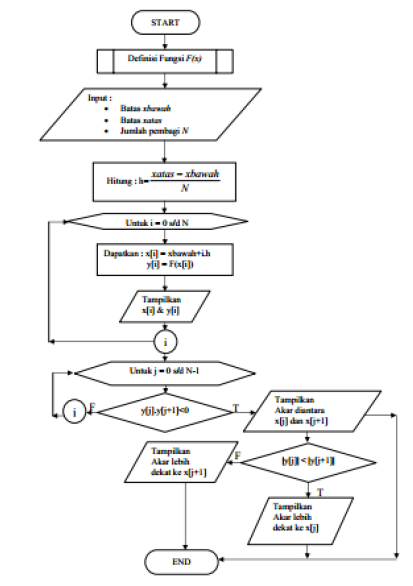
1. Algoritma
2. Definisikan fungsi f(x)
3. Tentukan range untuk x yang berupa batas bawah xbawah dan batas atas Xatas.
4. Tentukan jumlah pembagian N
5. Hitung step pembagi h



1. Untuk i = 0 s/d N dicari k dimana

* Bila f(Xk+1) = 0 maka Xk adalah penyelesaian
* Bila f(Xk+1) < 0 maka :
* Bila |f(Xk+1) maka Xk adalah penyelesaian
* Bila tidak Xk+1 adalah penyelesaian atau dapat dikatakan penyelesaian berada di antara Xk dan Xk+1

Flowchart



Listing Program

#include<stdio.h>

#include<math.h>

float a, b, n, e, xr;

int iterasi=0, kondisi=1;

void input();

void regulafalsi();

double f(double);

main()

{

char jawab;

do{

input();

regulafalsi();

fflush(stdin);

printf("\nMau menghitung lagi ? ");

scanf("%c", &jawab);

} while(jawab=='y' || jawab=='Y');

}

void input()

{

printf("\nMasukkan batas bawah : ");

scanf("%f", &a);

printf("Masukkan batas atas : ");

scanf("%f", &b);

printf("Masukkan jumlah iterasi maksimal : ");

scanf("%f", &n);

printf("Masukkan error : ");

scanf("%f", &e);

}

double f(double x)

{

return exp(-x)-x;

}

void regulafalsi()

{

printf("\ni\t a\t b\t\tx\t f(x)\t\tf(a)\n");

if(f(a)\*f(b)<0)

{

while(kondisi==1)

{

iterasi+=1;

xr=(f(b)\*a-f(a)\*b)/(f(b)-f(a));

printf("%d %f %f %f %f %f\n", iterasi, a, b, xr, f(xr), f(a));

if(fabs(f(xr))<e || iterasi>n)

kondisi=0;

else if(f(a)\*f(xr)<0){

b=xr;

}

else{

a=xr;

}

}

printf("\n\nNilai xr [Akar]\t= %lg\n", xr);

printf("f(xr) [Error]\t= %lg\n", f(xr));

}

else{

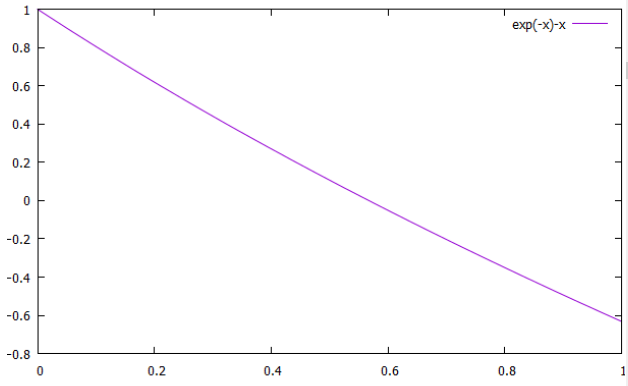
printf("Tidak ada akar");

}

printf("\n");

}

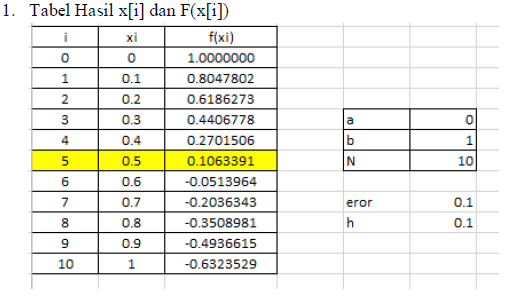
1. Pengamatan Awal
2. Gambar kurva fungsi F(x) = e-x-x dalam bentuk GNUPLOT



1. Perkiraan batas bawah dan batas atas

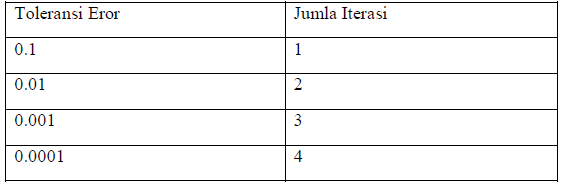
* Batas bawah = 0
* Batas atas = 1

1. Hasil Percobaan
2. Tabel Hasil x[i] dan F(x[i])



1. Pengamatan terhadap parameter
2. Toleransi error terhadap jumlah iterasi

* Tabel

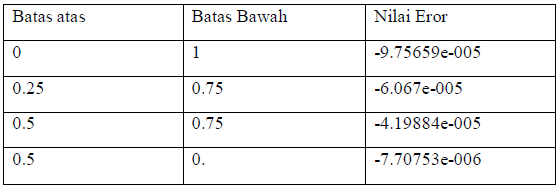


Analisa

* Dari percobaan diatas, dapat diketahui bahwa semakin banyak jumlah iterasi (N) maka besaran toleransi akan semakin mendekat kepada 0. Dengan jumlah iterasi yang besar, nilai pembagi area (h) akan semakin kecil sehingga toleransi eror semakin mendekati angka 0.

1. Pengubahan nilai batas atas dan batas bawah terhadap 20 iterasi

* Tabel



6

* Analisa

Semakin sempit range maka toleransi error semakin mendekati angka 0.

1. Kesimpulan

Setiap variabel yang dimasukkan ke percobaan mempengaruhi nilai toleransi error, baik jumlah iterasi, nilai batas atas, maupun batas bawah. Semakin banyak jumlah interasi maka jumlah toleransi eror akan semakin mendekati angka 0. Dan semakin kecil range antara batas atas dengan batas bawah, maka toleransi error akan semakin mendekati angka 0.