**University of Dhaka**

**Department of Computer Science and Engineering**

**CSE-3212: Numerical Methods Lab**

**3rd Year 2nd Semester**

**Session**: 2017 -18

**Name of the assignment:** Bisection Method

**Submitted by:**

Amit Roy, Roll: JH- 40

**Submitted to:**

Dr. Md.Haider Ali, Professor, Department of CSE,DU

Mr. Mubin Ul Haque,Lecturer, Department of CSE,DU

**Problem Statement:**

**1)** Given the following equation:

**f(x) = ex - 5x2 = 0**

1. Write a program which will find the value of **f(x)** where **x** is in the range **-1.0≤x≤1.0**. Increase the value of x by 0.1. Print the x and f(x) value in the console and also save the output in a **.csv file** and plot **f(x) vs x** graph from the .csv file.
2. In the same program, take two input **xlo**, **xhi** and **accuracy** where **f(xlo)** and **f(xhi)** have different sign. Use **Bisection Method** to find the root of the equation where relative approximate error is less than the accuracy.

**2)** Given the following equation:

**ln (osf) = -139.34411 +** **- + -**

1. Write a program which will find the value of **f(Ta)** where **Ta** = **t + 273.15** and t is in the range **0 ≤ t ≤ 40**. Increase the value of t by 1. Print **t**, **Ta** and **f**(**Ta**) value in the console and also save the output in a **.csv file** and plot **f(Ta) vs t** graph from the .csv file.
2. In the same program, take two input **tlo**, **thi** and **accuracy** where **f(tlo + 273.15)** and **f(thi + 273.15)** have different sign. Use **Bisection Method** to find the root of the equation where relative approximate error is less than the accuracy. Use 8,10,12 as the value of osf.

**Solution:**

**1)** Given equation:

**f(x) = ex - 5x2 = 0**

**Source code of solution:**

#include<stdio.h>

#include<math.h>

#include<stdlib.h>

#include<string.h>

#include<limits.h>

double lo,hi,accuracy;

FILE \*fp1,\*fp2;

double func(double x)

{

return (exp(x)-5.0\*x\*x);

}

double bisection\_method(double lo,double hi)

{

double prev,present,f,relative\_approx\_error;

int cnt = 1;

prev = (hi+lo)/2.0;

f = func(prev);

fprintf(fp2,"Iteration\tXl\tXu\tXr\tRelative Apporximate Error\n");

fprintf(fp2,"%d\t%0.6lf\t%0.6lf\t%0.6lf\n",cnt,lo,hi,prev);

cnt++;

if(!f) return prev;

else if(f \* func(hi) > 0) hi = prev;

else if(f \* func(lo) > 0) lo = prev;

while(1)

{

present = (hi+lo)/2.0;

f = func(present);

relative\_approx\_error = fabs((present-prev)\*100.00/present);

fprintf(fp2,"%d\t%0.6lf\t%0.6lf\t%0.6lf\t%0.7lf\n",cnt,lo,hi,present,relative\_approx\_error);

cnt++;

if(!f) return present;

else if(f \* func(hi) > 0) hi = present;

else if(f \* func(lo) > 0) lo = present;

if(relative\_approx\_error<accuracy) break;

prev = present;

}

return present;

}

int main()

{

fp1 = fopen("problem1\_1.csv","w");

fp2 = fopen("problem1\_2.csv","w");

printf("x\tF(x)\n");

fprintf(fp1,"x\tF(x)\n");

for(double i=-1.0;i<=1.0;i+=0.1)

{

printf("%0.2lf\t%0.6lf\n",i,func(i));

fprintf(fp1,"%0.2lf\t%0.6lf\n",i,func(i));

}

scanf("%lf %lf %lf",&lo,&hi,&accuracy);

double root = bisection\_method(lo,hi);  
 printf(“Root of the equation: %lf\n”,root);

fclose(fp1);

fclose(fp2);

}

**2)** Given equation:

**ln (osf) = -139.34411 +** **- + -**

**Source code of solution:**

#include<stdio.h>

#include<math.h>

#include<stdlib.h>

#include<string.h>

#include<limits.h>

#define osf 12 //changed file name for different osf value

double lo,hi,accuracy;

FILE \*fp1,\*fp2;

double func(double x)

{

return(-log(osf) - 139.34411 + (1.575701e5/x) - (6.642308e7/(x\*x)) + (1.2438e10/(x\*x\*x)) - ((8.621949e11)/(x\*x\*x\*x)));

}

double bisection\_method(double lo,double hi)

{

double prev,present,f,relative\_approx\_error;

int cnt = 1;

prev = (hi+lo)/2.0;

f = func(prev);

fprintf(fp2,"Iteration\tXl\tXu\tXr\tRelative Apporximate Error\n");

fprintf(fp2,"%d\t%0.6lf\t%0.6lf\t%0.6lf\n",cnt,lo-273.15,hi-273.15,prev-273.15);

cnt++;

if(!f) return prev;

else if(f \* func(hi) > 0) hi = prev;

else if(f \* func(lo) > 0) lo = prev;

while(1)

{

present = (hi+lo)/2.0;

f = func(present);

relative\_approx\_error = fabs((present-prev)\*100.00/present);

fprintf(fp2,"%d\t%0.6lf\t%0.6lf\t%0.6lf\t%0.7lf\n",cnt,lo-273.15,hi-273.15,present-273.15,relative\_approx\_error);

cnt++;

if(!f) return present;

else if(f \* func(hi) > 0) hi = present;

else if(f \* func(lo) > 0) lo = present;

if(relative\_approx\_error<accuracy) break;

prev = present;

}

return present-273.15;

}

int main()

{

fp1 = fopen("osf12\_1.csv","w");

fp2 = fopen("osf12\_2.csv","w");

fprintf(fp1,"t\tTa\tF(Ta)\n");

for(int i=0;i<=40;i++)

{

double x = i+273.15;

printf("%d\t%0.2lf\t%0.6lf\n",i,x,func(x));

fprintf(fp1,"%d\t%0.2lf\t%0.6lf\n",i,x,func(x));

}

scanf("%lf %lf %lf",&lo,&hi,&accuracy);

lo = lo + 273.15;

hi = hi + +273.15;

double root = bisection\_method(lo,hi);  
 printf(“Root of the equation: %0.6lf\n”,root);

fclose(fp1);

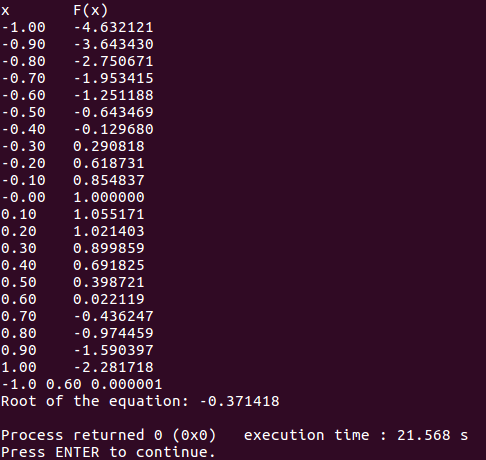
fclose(fp2);

}

**Sample Input/Output:**

**1)** Given equation:

**f(x) = ex - 5x2 = 0**

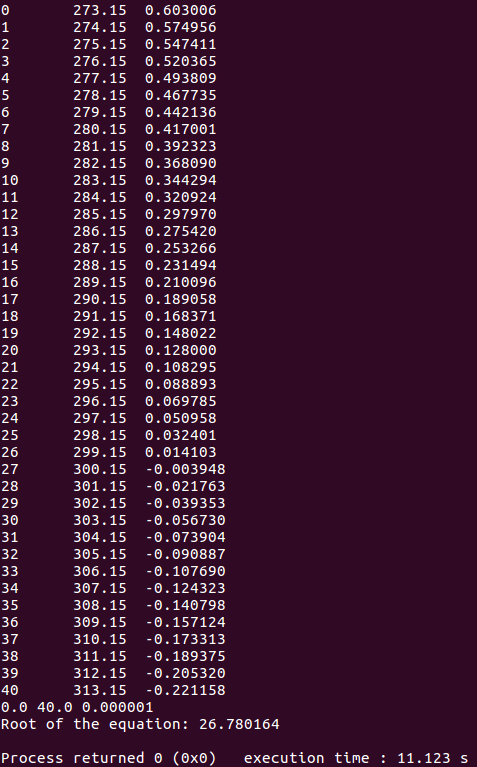


**Snapshot of Console**

**2)** Given equation:

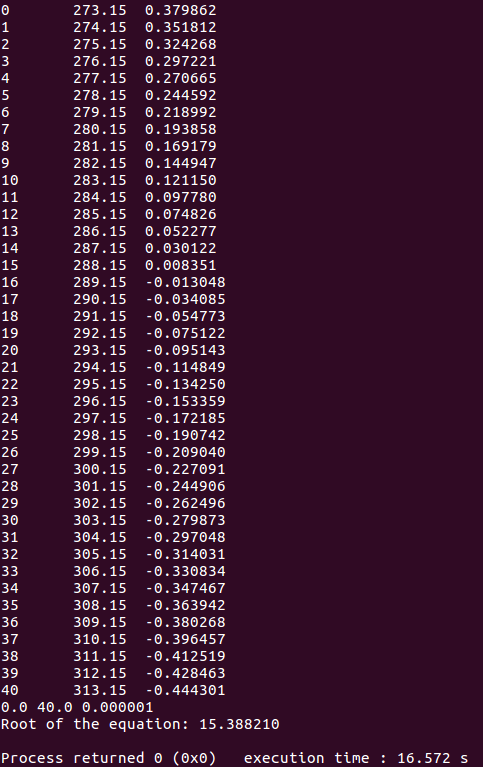
**ln (osf) = -139.34411 +** **- + -**

**OSF = 8:**

****

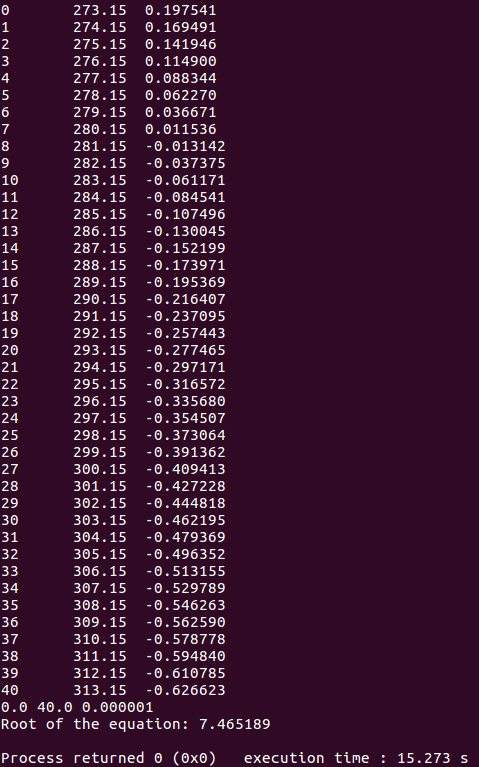
**Snapshot of Console**

**OSF = 10:**



**Snapshot of Console**

**OSF = 12:**

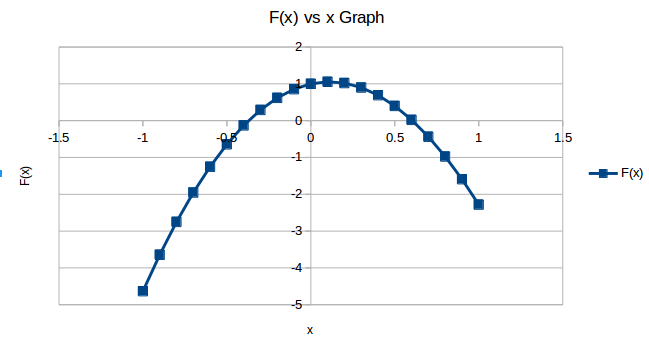
****

**Snapshot of Console**

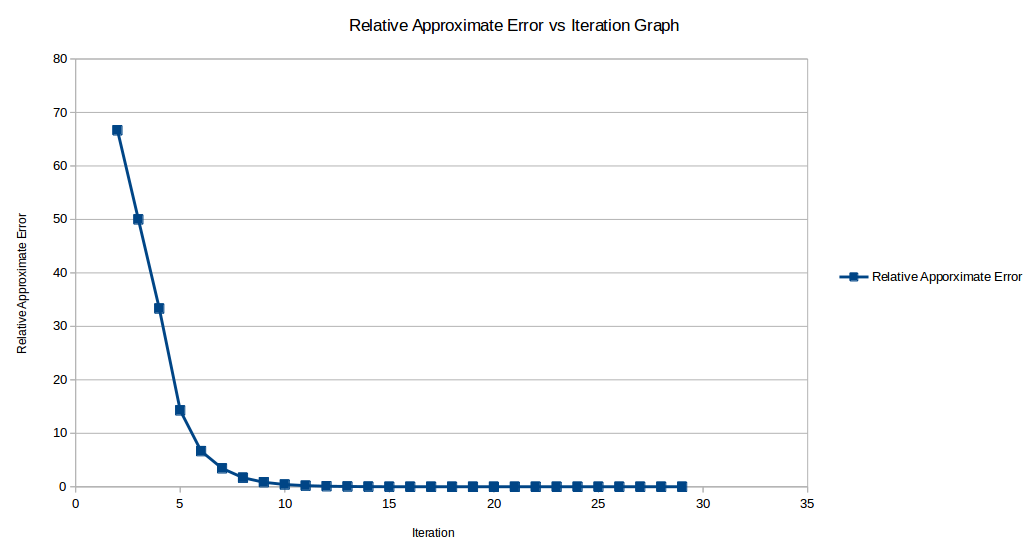
**Graphs:**

**1)** Given equation:

**f(x) = ex - 5x2 = 0**

****

**F(x) vs x Graph**

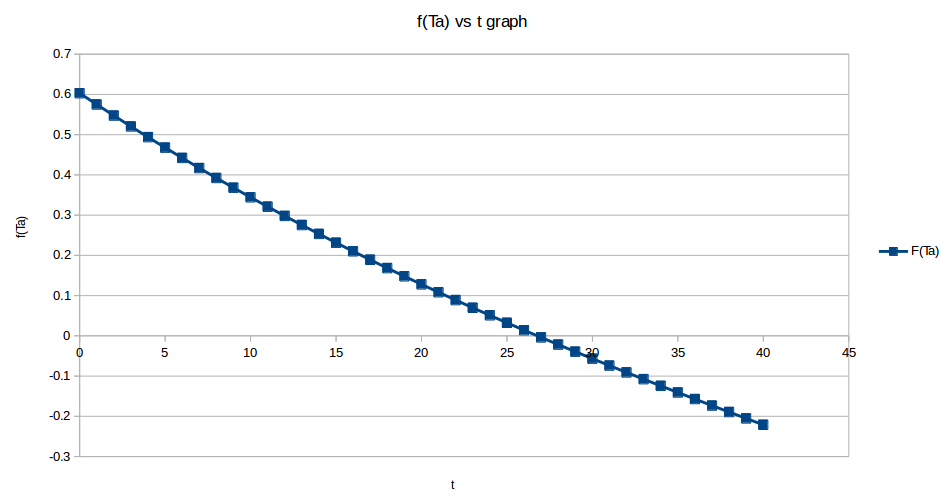
****

**Relative Approximate Error vs Iteration Graph**

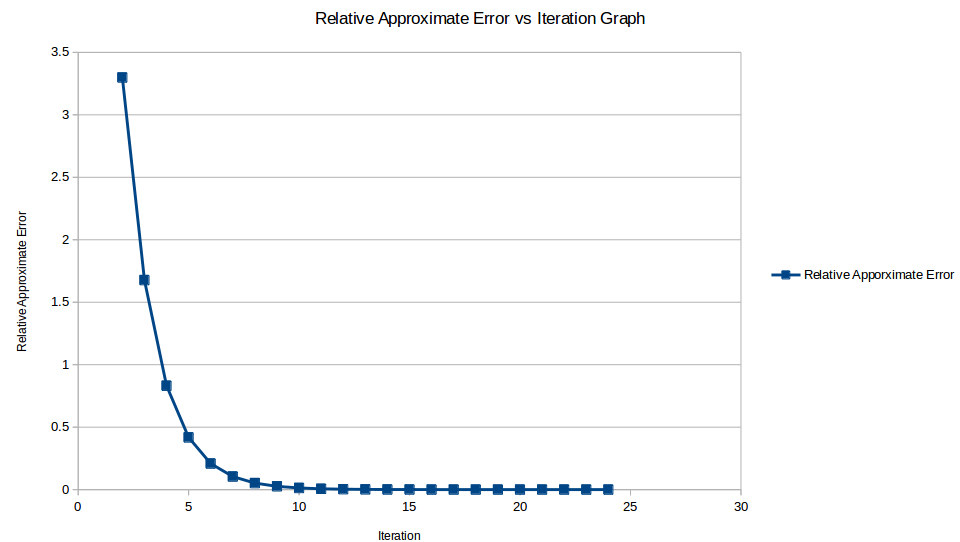
**2)** Given equation:

**ln (osf) = -139.34411 +** **- + -**

**OSF = 8:**

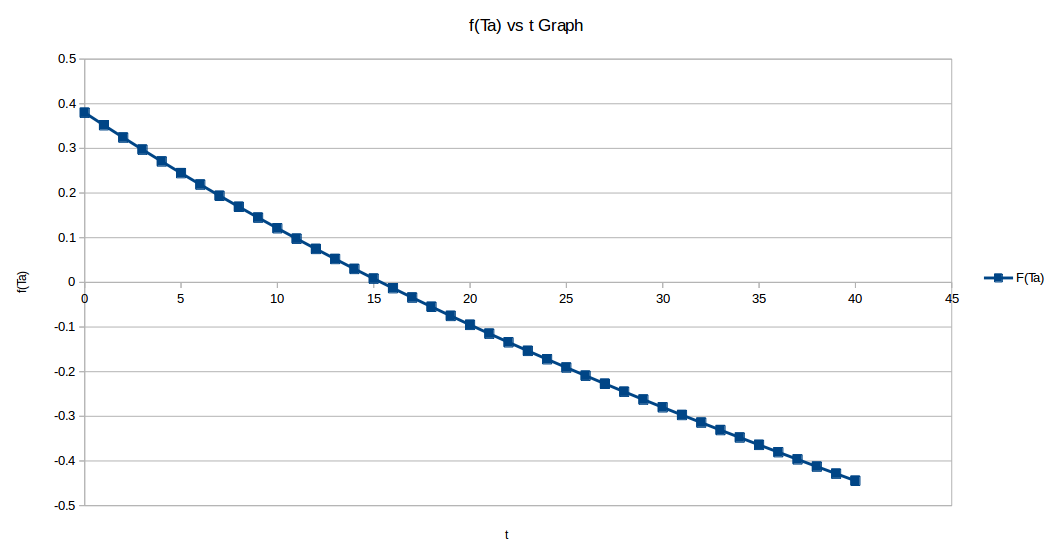
****

**f(Ta) vs t Graph**

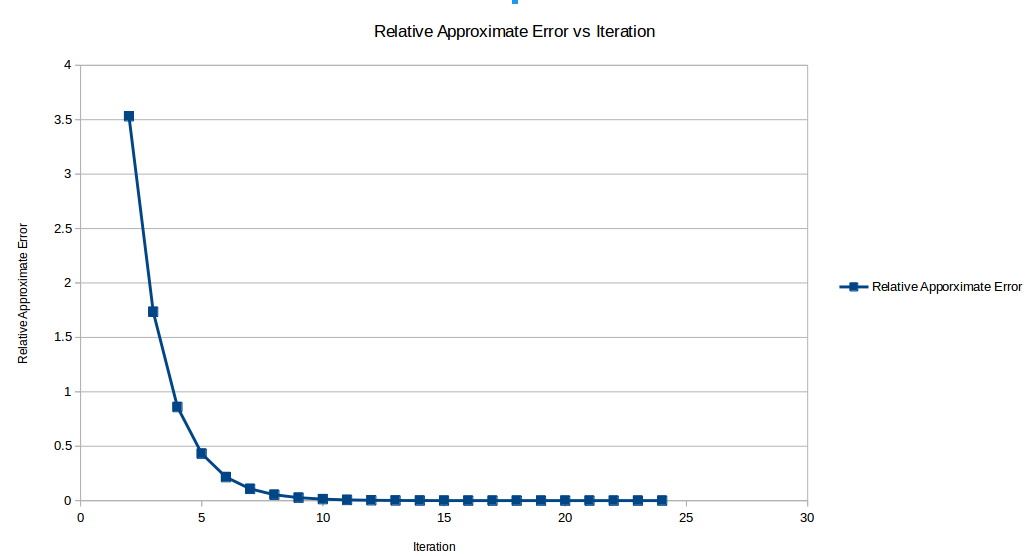
****

**Relative Approximate Error vs Iteration Graph**

**OSF = 10:**

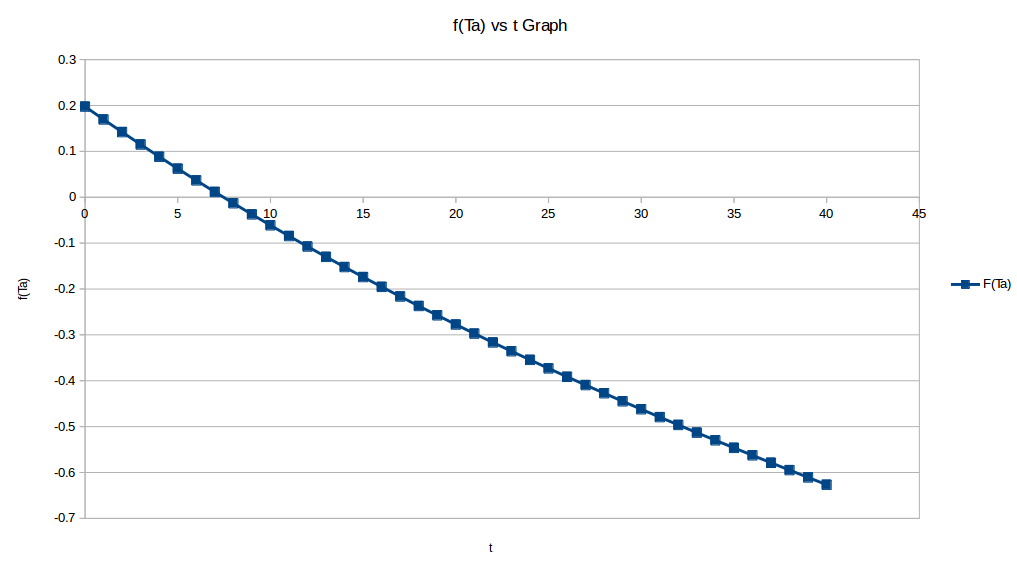
****

**f(Ta) vs t Graph**

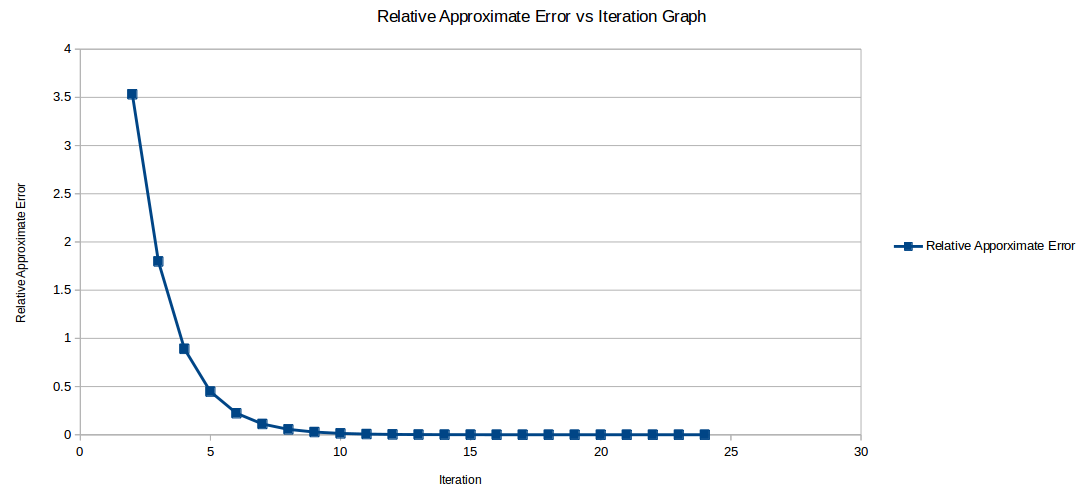
****

**Relative Approximate Error vs Iteration Graph**

**OSF = 12:**

****

**f(Ta) vs t Graph**

****

**Relative Approximate Error vs Iteration Graph**