Aishwary Sharma

2022CH11437

Group-1

Lab-5

f(x) = x – e^x/3

The solutions of the above equation are – 0.61906 and 1.512135.

1. **Bisection Method:**

// Header files and namespaces

#include<bits/stdc++.h>

using namespace std;

float f(float x){

return x - exp(x)/3 ;

}

void bisectionMethod(){

float xl, xr ;

cout << "What are the initial guesses- " ;

cin >> xl >> xr ;

float xi;

float error ;

int itr = 0;

if(f(xl)\*f(xr)> 0){

cout << "Initial guesses incorrect for Bisection Method (f(xl)\*f(xr)> 0)";

}

else{

while(true){

xi = (xl + xr)/2 ;

if(abs(xl-xi)<1e-04){

cout << "The solution comes out to be- " << xl << endl;

cout << "Number of iterations required- "<< itr << endl << endl;

return ;

}

else{

if(f(xl)\*f(xi)< 0){

xr = xi;

}

else{

xl = xi ;

}

itr ++ ;

}

}

}

}

int main() {

int i =2;

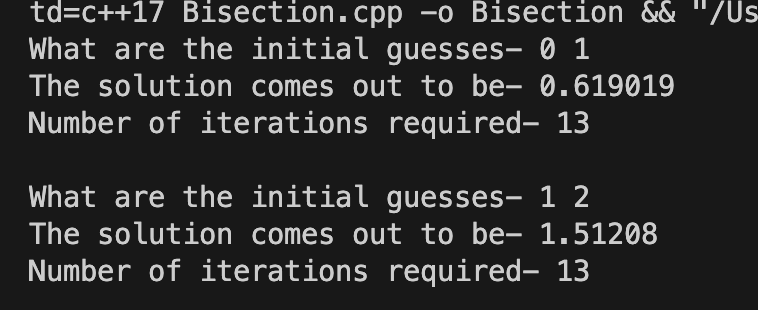
while(i--){

bisectionMethod();

}

}

**Output**:



1. **Regular Falsi Method**:

// Header files and namespaces

#include<bits/stdc++.h>

using namespace std;

float f(float x){

return x - exp(x)/3 ;

}

void RegularFalsi(){

float xl, xr ;

cout << "What are the initial guesses- " ;

cin >> xl >> xr ;

float xi;

float error ;

int itr = 0;

if(f(xl)\*f(xr)> 0){

cout << "Initial guesses incorrect for Bisection Method (f(xl)\*f(xr)> 0)";

}

else{

while(true){

xi = xl - f(xl)\*((xr-xl)/(f(xr)-f(xl))) ;

if(abs(xl-xi)<1e-04){

cout << "The solution comes out to be- " << xl << endl ;

cout << "Number of iterations required- " << itr << endl << endl ;

return ;

}

else{

if(f(xl)\*f(xi)< 0){

xr = xi;

}

else{

xl = xi ;

}

itr ++;

}

}

}

}

int main() {

int i =2;

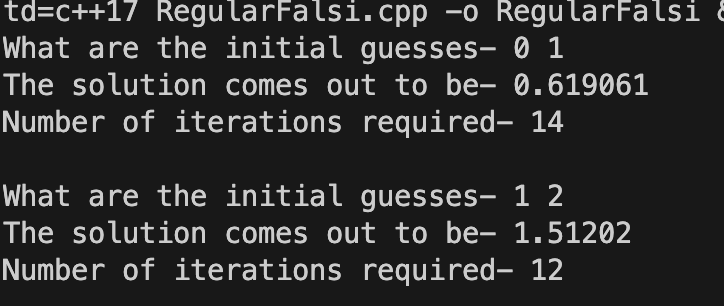
while(i--){

RegularFalsi();

}

}

**Output**:



1. **Secant Method:**

// Header files and namespaces

#include<bits/stdc++.h>

using namespace std;

float f(float x){

return x - exp(x)/3 ;

}

void Secant(){

float xl, xr ;

cout << "What are the initial guesses- " ;

cin >> xl >> xr ;

float xi;

float error ;

int itr = 0;

while(true){

xi = xl - f(xl)\*((xr-xl)/(f(xr)-f(xl))) ;

if(abs(xl-xi)<1e-04){

cout << "The solution comes out to be- " << xl << endl ;

cout << "Number of iterations required- " << itr << endl << endl;

return ;

}

else{

xl = xr ;

xr = xi;

itr ++;

}

}

}

int main() {

int i =2;

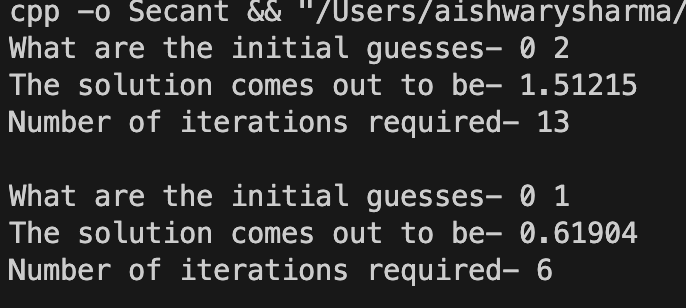
while(i--){

Secant();

}

}

**Output**:



1. **Fixed Point Iteration Method:**

// Header files and namespaces

#include<bits/stdc++.h>

using namespace std;

float g(float x){

return exp(x)/3 ;

}

// f(x)= x - e^x/3; x = g(x);

void FP(){

float x0 ;

cout << "What is the initial guess- " ;

cin >> x0 ;

float xi;

float error ;

int itr = 0;

while(true){

xi = g(x0);

if(abs(x0-xi)<1e-04){

cout << "The solution comes out to be- " << xi << endl ;

cout << "Number of iterations required- " << itr << endl << endl ;

return ;

}

else{

x0 = xi ;

itr ++;

}

}

}

int main() {

int i =2;

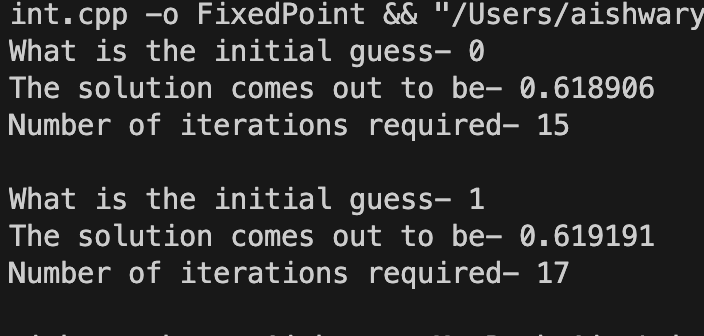
while(i--){

FP();

}

}

**Output:**



1. **Newton Raphson Method:**

// Header files and namespaces

#include<bits/stdc++.h>

using namespace std;

float f(float x){

return x - exp(x)/3 ;

}

float f1(float x){

return 1- exp(x)/3 ;

}

void NR(){

float x0 ;

cout << "What is the initial guess- " ;

cin >> x0 ;

float xi;

float error ;

int itr = 0;

while(true){

xi = x0 - f(x0)/f1(x0);

if(abs(x0-xi)<1e-04){

cout << "The solution comes out to be- " << xi << endl ;

cout << "Number of iterations required- " << itr << endl<< endl;

return ;

}

else{

x0 = xi ;

itr ++;

}

}

}

int main() {

int i =2;

while(i--){

NR();

}

}

**Output:**

