##### 

|  |  |
| --- | --- |
| Course Number | 1444 |
| Section Number | **1** |
| Course Title | MATHEMATICS FOR EMBEDDED SYSTEMS |
| Semester/Year | Summer/2019 |

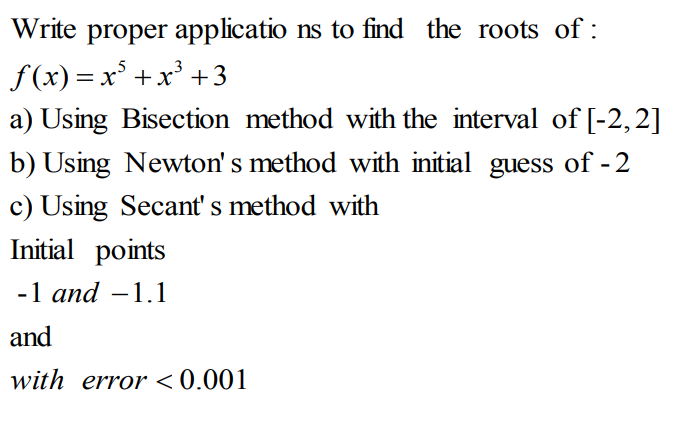
|  |  |
| --- | --- |
| Instructor | **Mohsen Salahi** |

|  |  |
| --- | --- |
| **Lab No.** | **5** |

|  |  |
| --- | --- |
| Submission Date | **25/07/2019** |
| Due Date | **28/07/2019** |

|  |  |  |  |
| --- | --- | --- | --- |
| Student Name | **Bibin Babu** | **Joyal Raju** | **Shinu Raj** |
| Student ID | C0749104 | C0749019 | C0749116 |
| Signature\* | **BB** | **JR** | **SR** |

*\*By signing above you attest that you have contributed to this submission and confirm that all work you have contributed to this submission is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a “0” on the work, an “F” in the course, or possibly more severe penalties.*



a)

#include<iostream>

using namespace std;

#define EPSILON 0.01

double func(double x)

{

return x\*x\*x\*x\*x + x\*x\*x + 3;

}

void bisection(double a, double b)

{

if (func(a) \* func(b) >= 0)

{

cout << "You have not assumed right a and b\n";

return;

}

double c = a;

while ((b-a) >= EPSILON)

{

c = (a+b)/2;

if (func(c) == 0.0)

break;

else if (func(c)\*func(a) < 0)

b = c;

else

a = c;

}

cout << "The value of root is : " << c;

}

int main()

{

double a =-2, b = 2;

bisection(a, b);

return 0;

}

OUTPUT



b)

#include<iostream>

#include<cmath>

#define EPSILON 0.001

using namespace std;

double func(double x)

{

return x\*x\*x\*x\*x + x\*x\*x + 3;

}

double derivFunc(double x)

{

return 5\*x\*x\*x\*x + 3\*x\*x;

}

void newtonRaphson(double x)

{

double h = func(x) / derivFunc(x);

while (abs(h) >= EPSILON)

{

h = func(x)/derivFunc(x);

x = x - h;

}

cout << "The value of the root is : " << x;

}

int main()

{

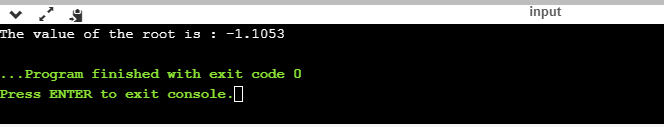
double x0 = -2;

newtonRaphson(x0);

return 0;

}

OUTPUT:



c)

#include <iostream>

#include<cmath>

using namespace std;

float f(float x)

{

float f = pow(x,5) + pow(x,3) +3;

return f;

}

void secant(float x1, float x2, float E)

{

float n = 0, xm, x0, c;

if (f(x1) \* f(x2) < 0) {

do {

x0 = (x1 \* f(x2) - x2 \* f(x1)) / (f(x2) - f(x1));

c = f(x1) \* f(x0);

x1 = x2;

x2 = x0;

n++;

if (c == 0)

break;

xm = (x1 \* f(x2) - x2 \* f(x1)) / (f(x2) - f(x1));

} while (fabs(xm - x0) <= E);

cout << "Root of the given equation=" << x0 << endl;

cout << "No. of iterations = " << n << endl;

} else

cout << "Can not find a root in the given inteval";

}

int main()

{

float x1 = -1, x2 = -1.1, E = 0.001;

secant(x1, x2, E);

return 0;

}

OUTPUT:

