Пермский национально исследовательский политехнический университет

Лабораторная работа “Численные методы решения уравнений”

Выполнила студентка группы РИС 23-3б:

Федорова Ольга Ильинична

Проверила доцент кафедры ИТАС:

Полякова Ольга Андреевна

Разработка алгоритма решения нелинейных уравнений.

Задача работы: решить нелинейное уравнение методом половинного деления и методом Ньютона (касательных).

Метод половинного деления

1. Анализ задачи.
2. Первый шаг: создание заданной функции. Она будет использоваться для вычисления нового, более приближенного к корню интервала.
3. Создание цикла while, который будет вычислять x до тех пор, пока его точность не будет меньше заданного эпсилона.
4. Поиск середины отрезка и проверка, является ли произведение функции от конца интервала и функции от середины отрезка меньше 0, чтобы уменьшить интервал поиска.
5. После завершения цикла выводится ответ.
6. Блок-схема программы:

0

1

{

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

c = (a+b)/2;

While (abs(b-a)>e){

#include <iostream>

#include <math.h>

using namespace std;

double f(double x) {

double f = 0.1\*pow(x,2)-x\*log(x);

return f; }

int main () {

stelocale(LC\_ALL, “Rus”)

double a=1, b=2, e=0.0000001, c;

}

}

{

}

{

b = c;

}

cout << (a + b) / 2 << endl;

a = c;

1. Код программы:

#include <iostream>

#include <math.h>

using namespace std;

double f(double x) {

double f = 0.1\*pow(x,2)-x\*log(x);

return f;

}

int main() {

setlocale(LC\_ALL, "Rus");

double a=1,b=2,e=0.0000001,c;

while (abs(b-a)>e) {

c = (a+b)/2;

if (f(b) \* f(c) < 0) {

a = c;

}

else {

b = c;

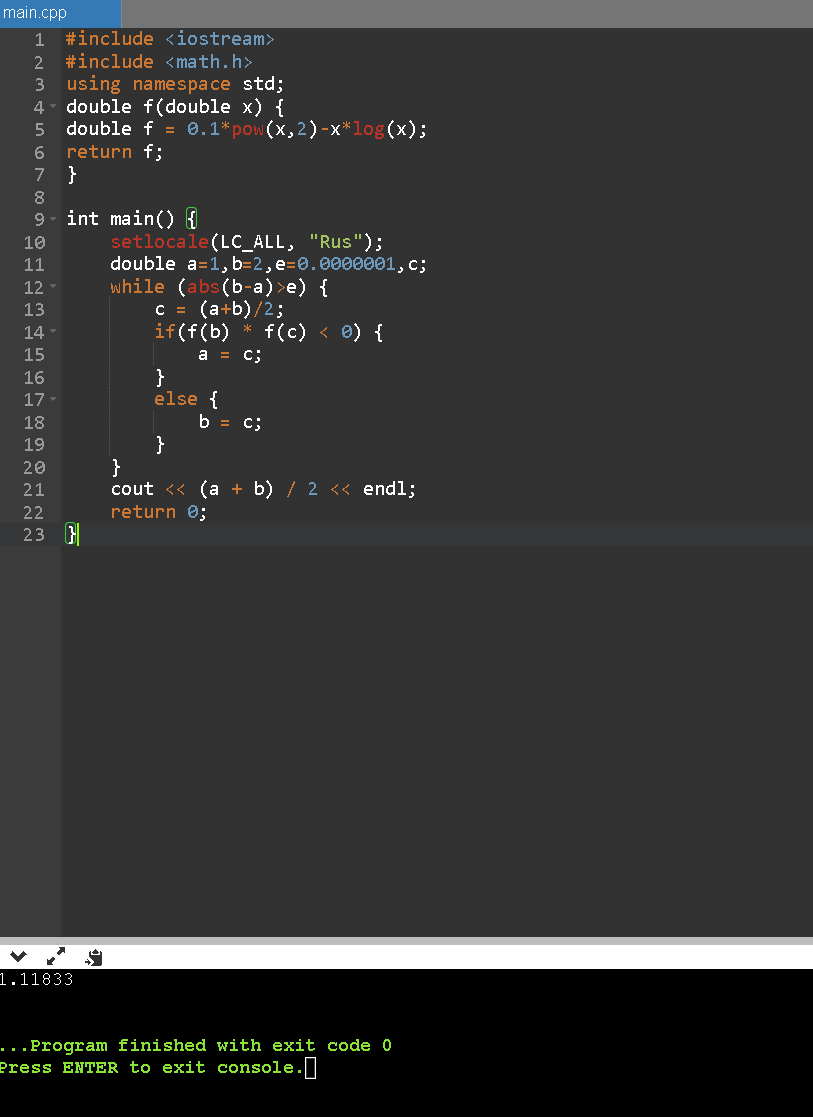
}

}

cout << (a + b) / 2 << endl;

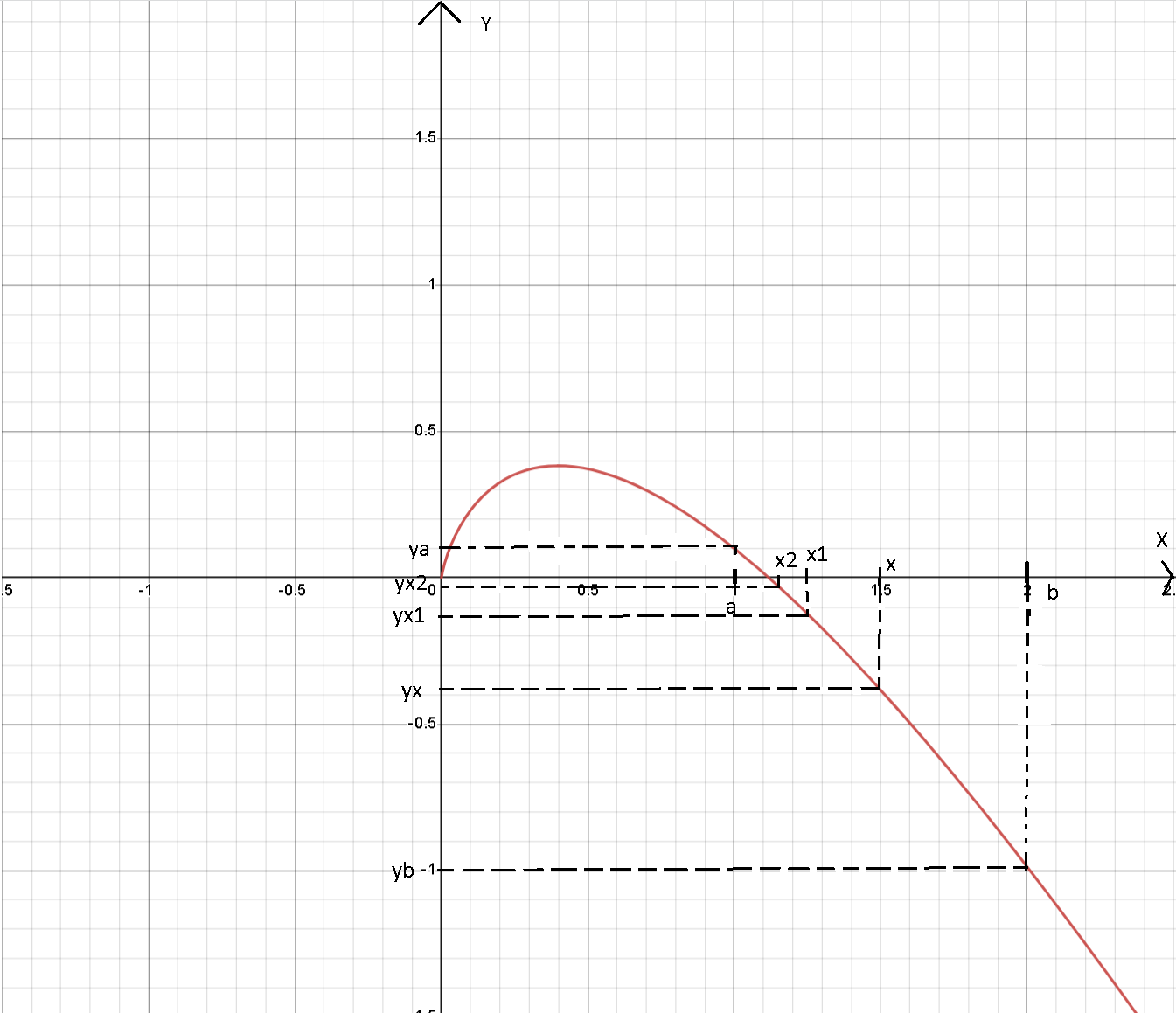
return 0;

}

1. Результат программы: 

Полученный ответ: 1.1183

1. Геометрическое решение:



Метод Ньютона(касательных)

1. Анализ задачи:
2. Создание двух функции: заданного уравнения и его производной. Благодаря им в дальнейшем будет высчитываться новый приближенный к ответу х.
3. Проверка, от какого конца интервала лучше начинать двигаться.
4. х1 – переменная, которая будет запоминать n-1-ое значение х.
5. Цикл do while, который первым шагом рассчитает первый х, полученный по

формуле, а далее будет повторяться и сравнивать значение |x-x1| b эпсилон, пока эпсилон не станет больше.

1. После завершения цикла выводится ответ.
2. Блок-схема программы:

#include <iostream>

#include <cmath>

using namespace std;

double f(double x) {

double f = 0.1\*pow(x,2)-x\*log(x);

return f; }

double pr(double x) {

double pr = 0.2\*x-log(x)-1;

return pr; }

int main() {

setlocale(LC\_ALL, "Rus");

double a=1,b=2,e=0.00000001,x0,x,x1;

if (f(a)>0)

1 0

{ {

x0 = a;

x0 = b;

} }

x = x0;

cout<<x<<endl;

while (abs(x-x1)>e);

x = x-f(x)/pr(x);

x1 = x;

do

{

}

0

1

}

1. Код программы:

#include <iostream>

#include <cmath>

using namespace std;

double f(double x) {

double f = 0.1\*pow(x,2)-x\*log(x);

return f;

}

double pr(double x) {

double pr = 0.2\*x-log(x)-1;

return pr;

}

int main() {

setlocale(LC\_ALL, "Rus");

double a=1,b=2,e=0.00000001,x0,x,x1;

if (f(a)>0) {

x0 = a;

}

else {

x0 = b;

}

x = x0;

do {

x1 = x;

x = x-f(x)/pr(x);

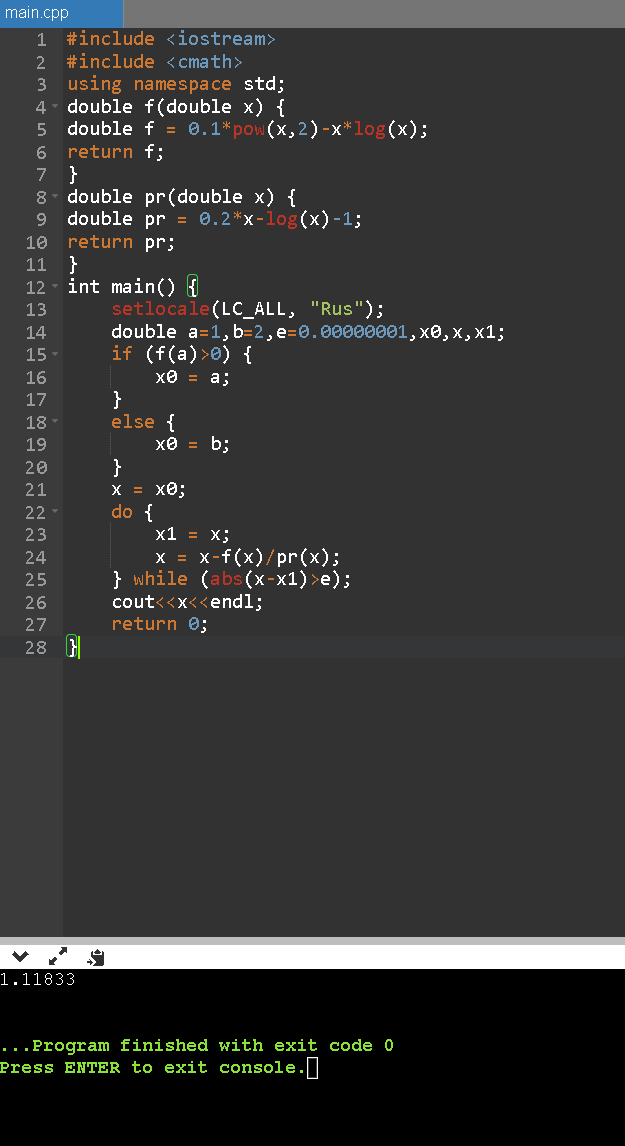
} while (abs(x-x1)>e);

cout<<x<<endl;

return 0;

}

1. Результат программы:



Полученный ответ: 1.11833

1. Геометрическое решение:

