**Отчет**

**по Методам численного анализа**

Исполнитель

студент 161 группы

специальности

Прикладная математика

Борис Д. Ю.

«9» Марта 2018 г.

Гродно, 2018

**Лабораторная работа №2**

Тема: **Численное решение нелинейных уравнений**

**(метод хорд, метод Эйткена, метод Стеффенсона )**

1. **Постановка задачи**

При помощи указанных методов узнать приближённое значение корней и количество итераций для следующей функции:

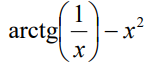
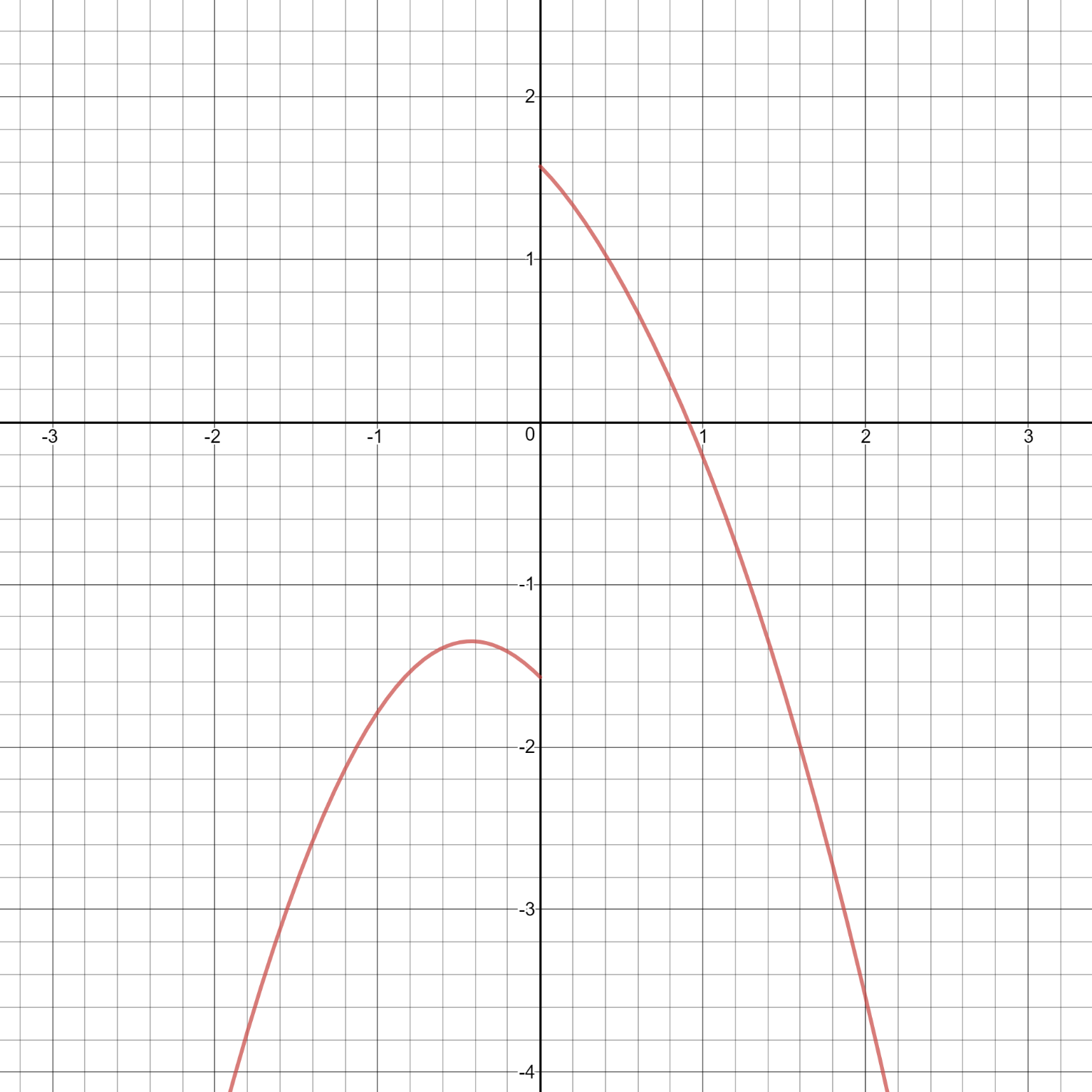


График этой функции: 

1. **Теоретический материал**

*Алгоритм метода хорд:*  
Из двух отрезков [*a*, *x*0] и [*x*0, *b*] выберем тот, на концах которого функция *f*(*x*) имеет разные знаки (т. е. отрезок, содержащий корень уравнения). Проведем секущую на новом отрезке [*x*0, *b*] и получим следующее приближение корня *x*1. Формируем последовательность *x*0, *x*1, *x*2, …, *x*n, … до выполнения условия

*Aлгоритм**метода Эйткена*

1. Берем x0 – начального приближения

2. Вычисляем значений x1=ϕ(x0) x2=ϕ (x1).

3. Находим x2~ по формуле: x2~=(x0\*x2-x2\*x2)/(x2-2x1+x0).

4. Вычисляем: x3= ϕ(x2~).

5. Проверка на точность: |f(x3)| ≤ ε.

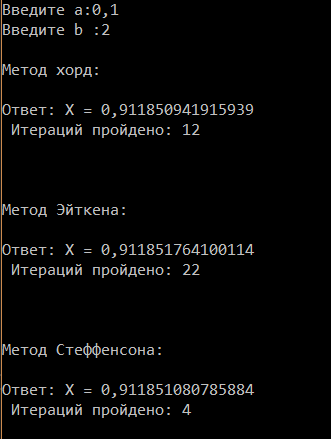
6. Условие не выполнено. Полагаем x0=x2~,x1=x3 вычисляем x2=ϕ(x1)и вернуться к шагу 3.

*Aлгоритм метода Стеффенсона*

Используя метод итерации и метод Эйткена можно получить новую итерационную формулу для нахождения приближенного корня уравнения. Итерационная формула Стеффенсона

1. **Результат вычислений**

Результат вычислений для отрезка [0.1;2]:



P.S. не забываем что делить на 0 нельзя.

1. **Текст программы**

*Метод хорд*

public static void method\_chord(double a, double b, double e)

{

Console.WriteLine("\nМетод хорд:\n");

double x\_next = 0;

double tmp;

int i = 3;

do

{

tmp = x\_next;

x\_next = b - F(b) \* (a - b) / (F(a) -F(b));

a = b;

b = tmp;

i++;

} while (Math.Abs(x\_next - b) > e);

Console.WriteLine("Ответ: X = {0}\n Итераций пройдено: {1}\n\n", x\_next, i);

}

*Метод Эйткена*

public static void method\_aitken(double a, double b,double e)

{

Console.WriteLine("\nМетод Эйткена:\n");

double x0 = (a + b) / 2.0;

double x1 = func(x0);

double x2 = func(x1);

double x\_temp = (x0 \* x2 - (x1\*x1)) / (x2 - 2 \* x1 + x0);

double x3 = func(x\_temp);

int i = 1;

while (Math.Abs(F(x3)) > e)

{

x0 = x\_temp;

x2 = func(x1);

x1 = x3;

x\_temp= (x0 \* x2 - (x1 \* x1)) / (x2 - 2 \* x1 + x0);

x3 = func(x\_temp);

i++;

}

Console.WriteLine("Ответ: X = {0}\n Итераций пройдено: {1}\n\n", x\_temp, i);

}

*Метод Стеффенсона*

public static void method\_stephenson(double a, double b, double e)

{

Console.WriteLine("\nМетод Стеффенсона:\n");

double x = (a + b) / 2.0;

int i = 1;

while (Math.Abs(F(x))>e)

{

x = x\_n(x);

i++;

}

Console.WriteLine("Ответ: X = {0}\n Итераций пройдено: {1}\n\n", x, i);

}