Московский авиационный институт

(национально-исследовательский университет)

Институт информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

КУРСОВОЙ ПРОЕКТ

по курсу

“Вычислительные системы”

I семестр

“Процедуры и функции в качестве параметров”

Студент: Почечура А. А.

Группа: М8О-106Б-20

Руководитель: Дубинин Алексей Владимирович

Оценка:

Дата:

Москва, 2020

**Оглавление**

[1 Введение 3](#_Toc59480930)

[2 Задание 4](#_Toc59480931)

[3 Метод дихотомии (половинного деления). 5](#_Toc59480932)

[4 Метод итераций 7](#_Toc59480933)

[5 Метод Ньютона 11](#_Toc59480934)

[6 Описание программы 13](#_Toc59480935)

[7 Выводы 14](#_Toc59480936)

[8 Список литературы 14](#_Toc59480937)

**1. Введение**

Важным при написании программного кода является не только факт его работоспособности, но и его общая структура. Во многом благодаря функциям мы можем повторно использовать куски кода, чтобы многократно производить однотипные операции. При этом иногда функция должна работать с другой функцией, в этом случае последнюю можно передать в качестве параметра, что будет рассмотрено в данной работе.

Кроме того, рассмотрим различные численные способы решения уравнений и сравним их по следующим параметрам: точность вычисления относительно друг друга и количество итераций, произведенных для вычисления корня.

**2. Задание**

Составить программу на языке СИ с процедурами решения трансцендентных алгебраических уравнений различными численными методами (итераций, Ньютона и половинного деления - дихотомии). Нелинейные уравнения оформить как параметры-функции, разрешив относительно неизвестной величины в случае необходимости. Применить каждую процедуру к решению двух уравнений, заданных двумя стоками таблицы, начиная с варианта с заданным номером. Если метод неприменим, дать математическое обоснование и графическую иллюстрацию.

Варианты заданий: 15,16.

Уравнение 15: ;

Отрезок, содержащий корень: [1,2];

Приближённое значение: 1,2388.

Уравнение 16: ;

Отрезок, содержащий корень: [2,3];

Приближённое значение: 2,2985.

# **3. Метод дихотомии (половинного деления)**

**Метод дихотомии** или метод деления [отрезка](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D1%80%D0%B5%D0%B7%D0%BE%D0%BA) пополам — простейший численный метод для решения [нелинейных уравнений](https://ru.wikipedia.org/wiki/%D0%9D%D0%B5%D0%BB%D0%B8%D0%BD%D0%B5%D0%B9%D0%BD%D1%8B%D0%B5_%D1%83%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D1%8F) вида f(x)=0.

Если мы ищем ноль, то на концах отрезка функция должна быть противоположных знаков. Разделим отрезок пополам и возьмём ту из половинок, на концах которой функция по-прежнему принимает значения противоположных знаков. Если значение функции в серединной точке оказалось искомым нулём, то процесс завершается.

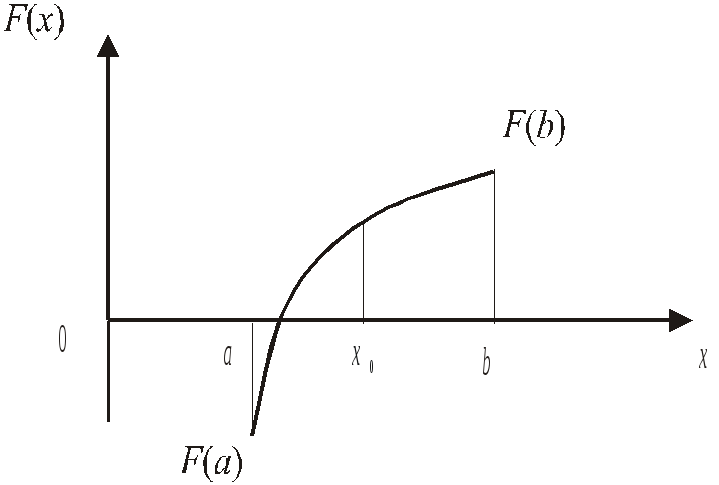
Пусть:

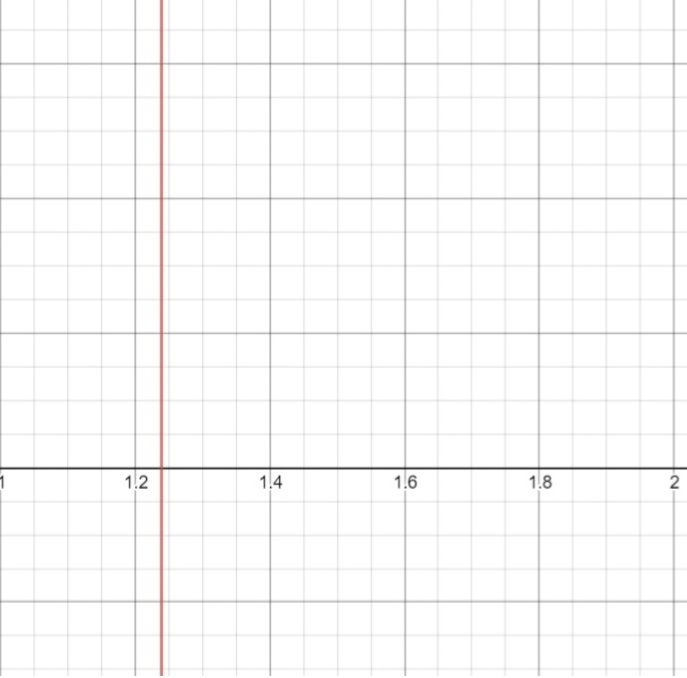
* xn — начало отрезка по х;
* xk — конец отрезка по х;
* xi — середина отрезка по х;
* eps — требуемая точность вычислений;

Условие окончания: xk - xn < eps;

Итерационный процесс: если f(xn)\*f(xi)<0, то xk = xi, иначе xn = xi;

Приближённое значение корня: (xk + xn)/2.





1. Окрестность корня уравнения .



1. Окрестность корня уравнения .

# **4. Метод итераций**

**Метод итераций** — один из простейших численных методов решения

[уравнений](https://ru.wikipedia.org/wiki/%D0%A3%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5). Метод основан на принципе [сжимающего отображения](https://ru.wikipedia.org/wiki/%D0%A1%D0%B6%D0%B8%D0%BC%D0%B0%D1%8E%D1%89%D0%B5%D0%B5_%D0%BE%D1%82%D0%BE%D0%B1%D1%80%D0%B0%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5), который применительно к численным методам в общем виде также может называться методом простой итерации или методом последовательных приближений.

Идея метода простой итерации состоит в том, чтобы уравнение f(x)=0 привести к эквивалентному уравнению φ(x)=x, так, чтобы отображение φ(x) было сжимающим. Если это удаётся, то последовательность итераций

сходится. Такое преобразование можно делать разными способами. В частности, сохраняет корни уравнение вида φ(x) = x – λ(x)\* f(x),

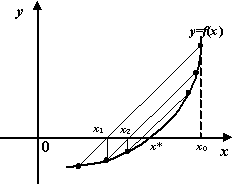
если λ(x) 0 на исследуемом отрезке. Оптимальным выбором является

λ(x) = , что приводит к [методу Ньютона](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_%D0%9D%D1%8C%D1%8E%D1%82%D0%BE%D0%BD%D0%B0), который является быстрым, но требует вычисления производной.

Формула итераций оказывается предельно простой: и на каждой итерации нужно один раз вычислить значение функции f(x).

Эта формула легко выводятся из геометрических соображений. Рассмотрим прямую, проходящую через точку (;) на графике y= с угловым коэффициентом . Тогда уравнением этой прямой будет .

{\displaystyle \displaystyle y=f(x\_{i})+{\frac {1}{{\lambda }\_{0}}}(x-x\_{i}).}Найдём точку пересечения этой прямой с осью Ox из уравнения: {\displaystyle \displaystyle f(x\_{i})+{\dfrac {1}{{\lambda }\_{0}}}(x-x\_{i})=0,}откуда {\displaystyle x=x\_{i}-{\lambda }\_{0}f(x\_{i})=x\_{i+1}}. Следовательно, эта прямая пересекает ось Ox как раз в точке следующего приближения.



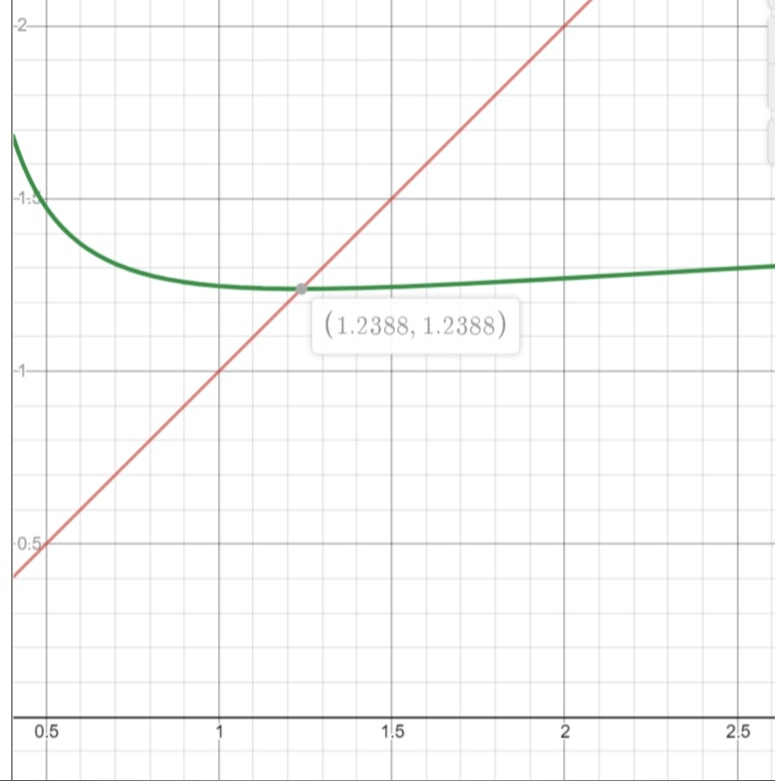
Начальное приближение корня: , где a и b – начало и конец отрезка;

Условие сходимости метода: ;

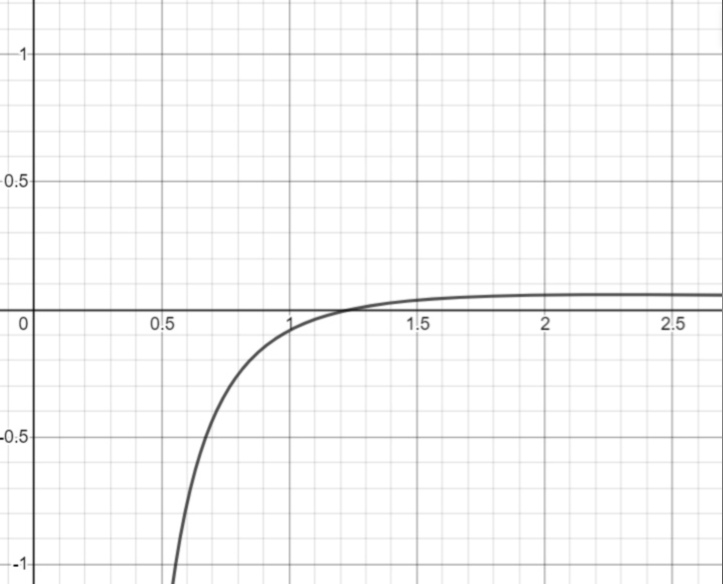
Итерационный процесс: ;

Условие окончания: ;

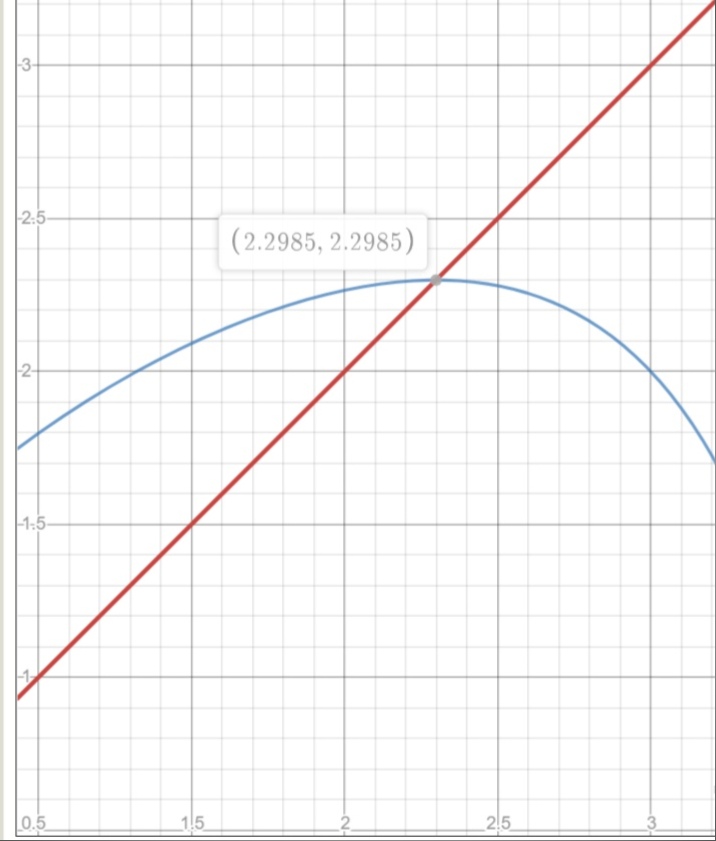
Приближённое значение корня: .



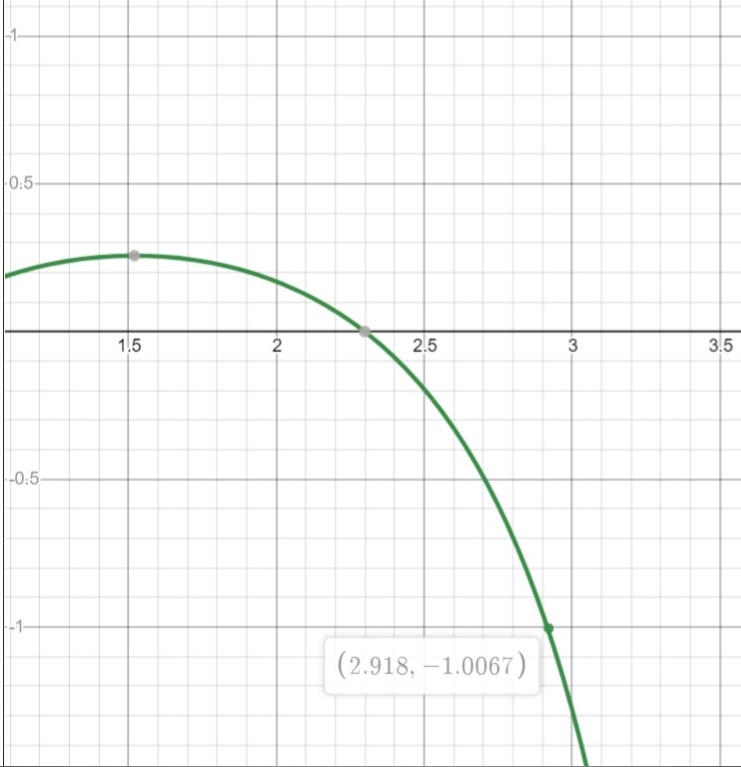
1. График и φ(x) = x – .



1. График φ(x) = (условие сходимости).



1. График и .

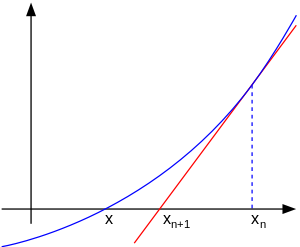


1. График φ(x) = (условие сходимости).

# **5. Метод Ньютона**

**Метод Ньютона** (также известный как метод касательных) — это итерационный [численный метод](https://ru.wikipedia.org/wiki/%D0%A7%D0%B8%D1%81%D0%BB%D0%B5%D0%BD%D0%BD%D1%8B%D0%B5_%D0%BC%D0%B5%D1%82%D0%BE%D0%B4%D1%8B) нахождения корня ([нуля](https://ru.wikipedia.org/wiki/%D0%9D%D1%83%D0%BB%D1%8C_%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%B8)) заданной функции.

Основная идея метода заключается в следующем: задаётся начальное приближение вблизи предположительного корня, после чего строится касательная к графику исследуемой функции в точке приближения, для которой находится пересечение с осью абсцисс. Эта точка берётся в качестве следующего приближения. И так далее, пока не будет достигнута необходимая точность.



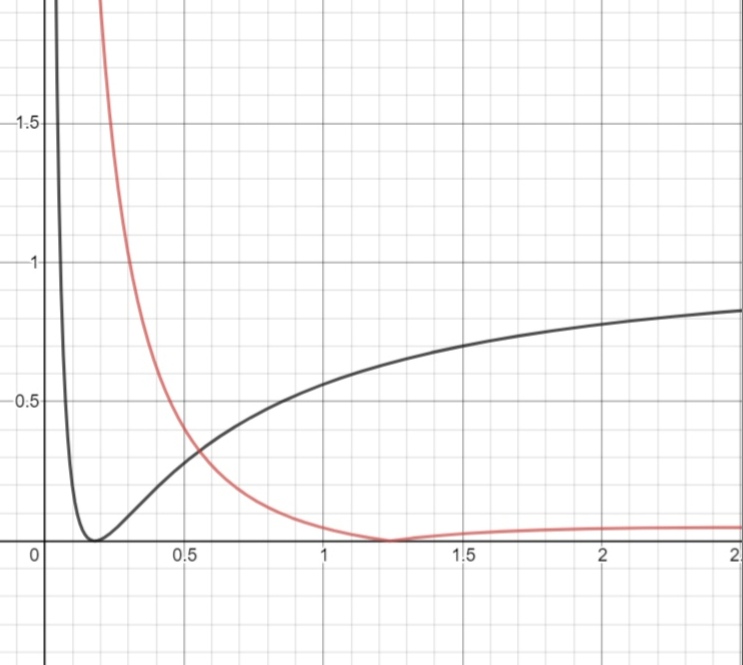
Начальное приближение корня: , где a и b – начало и конец отрезка;

Условие сходимости метода: ;

Итерационный процесс: ;

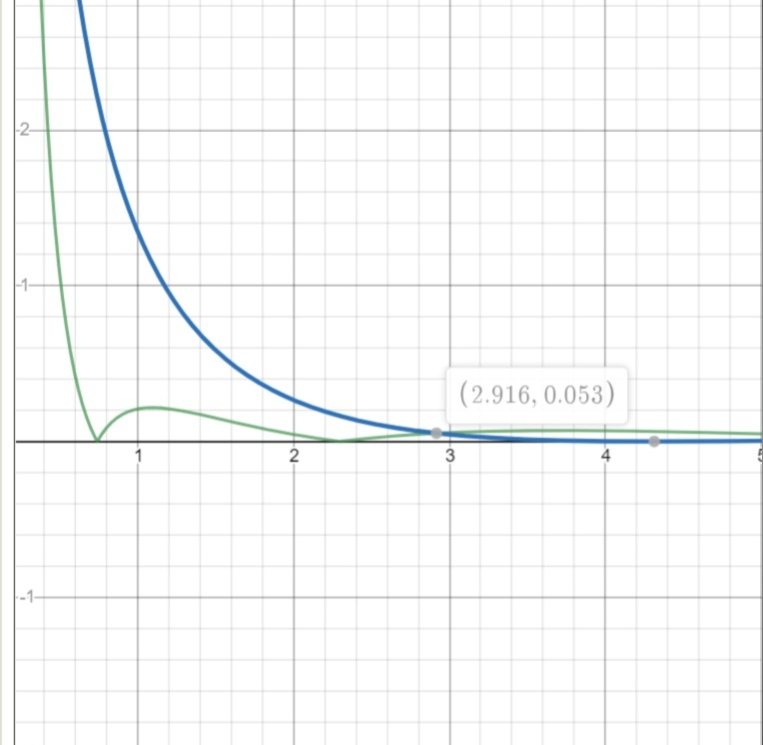
Условие окончания: ;

Приближённое значение корня: .



1. Чёрный график: ;

Красный график: .



1. Синий график: ;

Зелёный график: .

# **6. Описание программы**

Сперва создадим два typedef’а: typedef double(\*function)(double), чтобы можно было удобно передавать данные нам функции в качестве аргументов другим функциям, и typedef struct result, с помощью которого мы будем создавать переменные и хранить в них нужные нам значения для получения ответа. Затем напишем функцию вычисления машинного эпсилона и зададим эпсилон как глобальную переменную. Потом напишем функции уравнений и приступим к осуществлению каждого из рассмотренных выше методов. Для метода дихотомии дополнительных функций не потребуется, для метода Ньютона нужно написать функцию первой и второй производной, для метода итераций потребуется написать функцию φ(x) и отдельно производную для неё. Считаем delta=e\*k, задавая k отдельно. Рассчитываем нужные нам значения.

# **7. Выводы**

Использование структурных типов в качестве возвращаемого значения функции позволяет возвращать больше одного значения, что делает программу более модульной и компактной. А передача математических функций в качестве параметра другим функциям позволяет повторно использовать данные функции для разных математических уравнений, что упрощает код самой программы.

Все методы оказались применимы и их результаты совпадают с точностью до машинного эпсилон.

# **8. Список литературы**

1. [Метод бисекции — Википедия (wikipedia.org)](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_%D0%B1%D0%B8%D1%81%D0%B5%D0%BA%D1%86%D0%B8%D0%B8).
2. [Метод простой итерации — Википедия (wikipedia.org)](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_%D0%BF%D1%80%D0%BE%D1%81%D1%82%D0%BE%D0%B9_%D0%B8%D1%82%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%B8).
3. [Метод Ньютона — Википедия (wikipedia.org)](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_%D0%9D%D1%8C%D1%8E%D1%82%D0%BE%D0%BD%D0%B0).