Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Пермский национальный исследовательский

политехнический университет»

Кафедра “Информационные технологии и автоматизированные системы”

**О Т Ч Ё Т**

**по Лабораторной работе №1 вариант 6**

Дисциплина: основы алгоритмизации и программирования

Тема: Методы решения нелинейных уравнений.

Выполнил работу

Студент группы ИВТ-24-2б

Роговик.Н.А

Проверила

Доцент кафедры ИТАС

Полякова.О.А

Пермь, 2024

Цель и задачи

Цель – научиться составлять блок-схемы и программные коды следующих методов – Итераций, Ньютона, Половинного деления

Для достижения поставленной цели необходимо решить следующие задачи:

Решить уравнение - 0,25x3 + x - 1,2502 = 0, методом Итераций, методом Ньютона, методом Половинного деления.

Метод Ньютона

**Условие задачи**: решить методом Ньютона уравнение - 0,25x3 + x - 1,2502 = 0 отрезок, содержащий корень: [0;2], точное значение: 1,0001

**Анализ задачи**: Написать код и составить блок-схему уравнения - 0,25x3 + x - 1,2502 = 0, на C++ и решить методом ньютона

**Метод Ньютона:**

Задаем начальное приближение x0 (можно выбрать любое значение из отрезка [0, 2]).

Выполняем итерации метода:

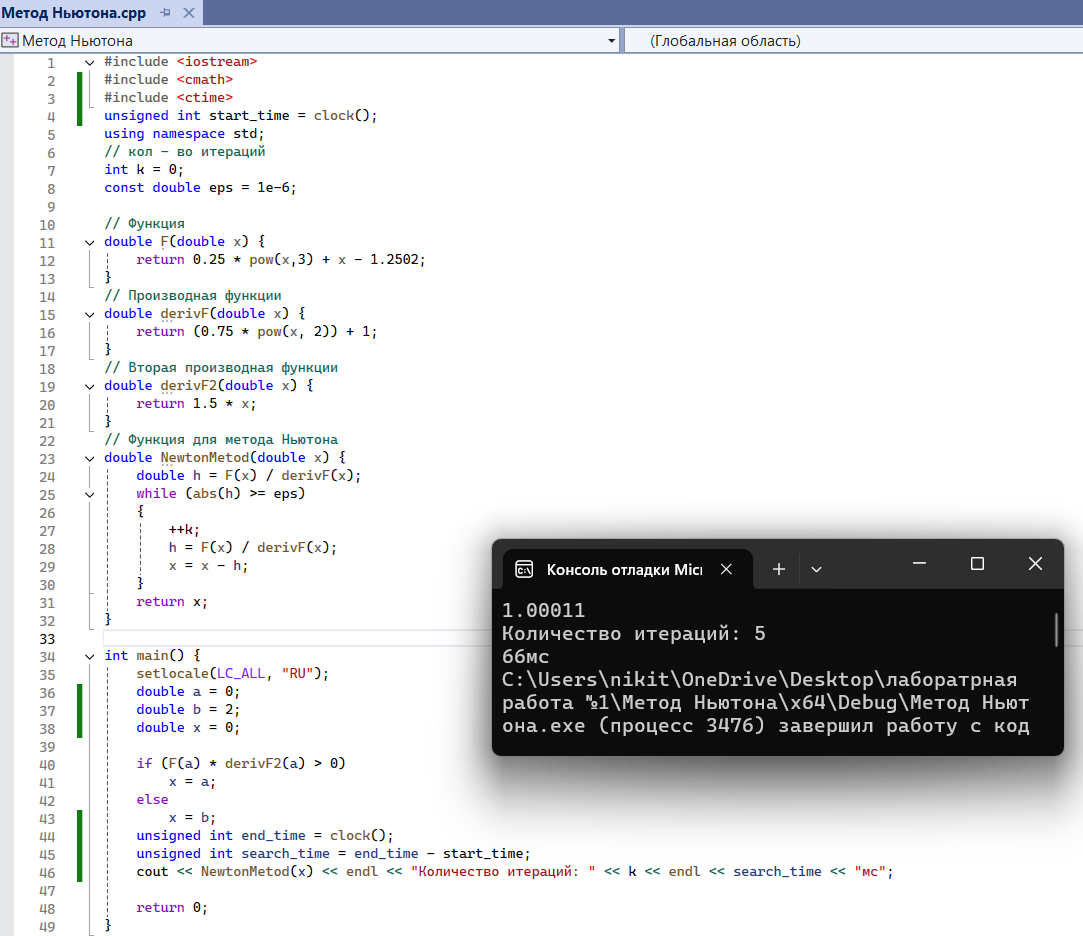
На каждой итерации вычисляем значение функции и ее производной в точке x: f(x) и f '(x).

Обновляем значение x по формуле: x = x - f(x) / f '(x).

Проверяем условие |x – x-1| < epsilon. Если условие выполняется, выводим значение x как приближенный корень.

Иначе, возвращаемся к шагу 2 и продолжаем итерации до достижения заданной точности или нахождения корня.

Код C++:



Блок-схема:

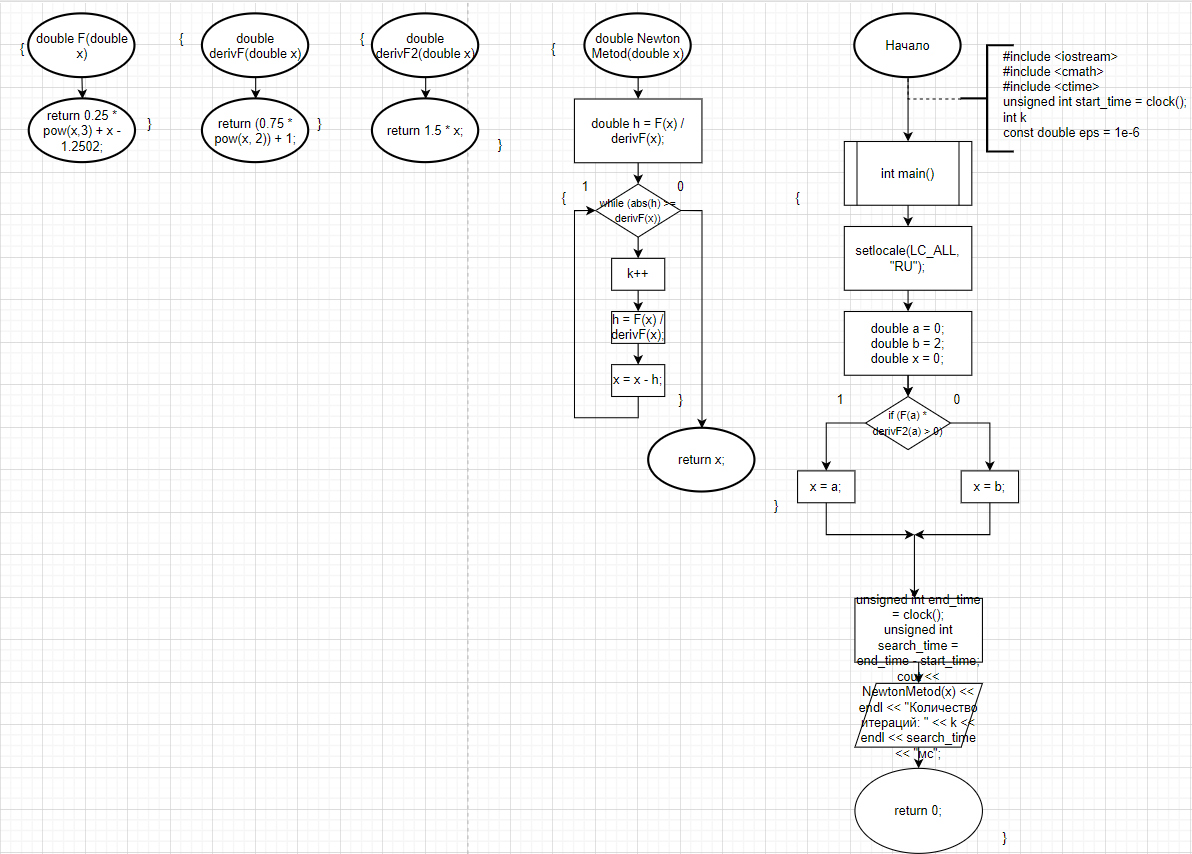
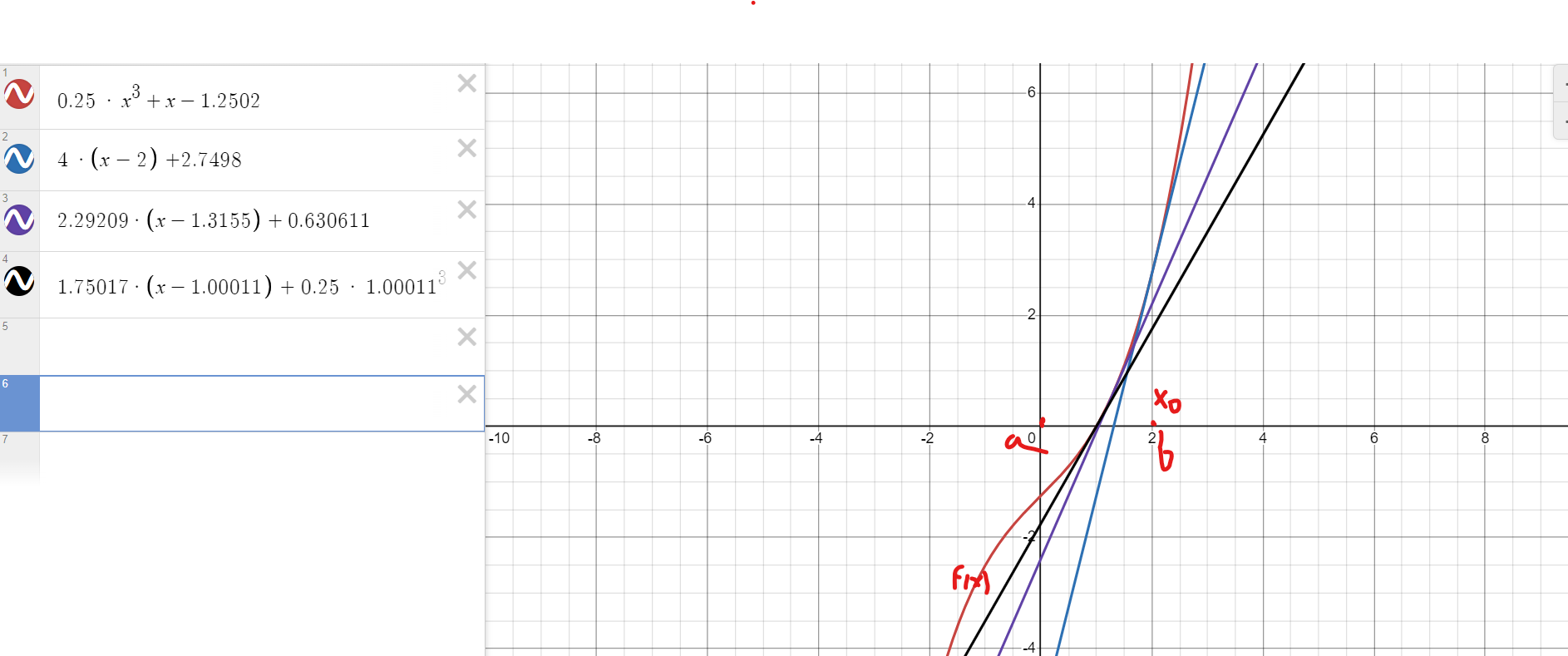


График:



Метод Половинного деления

**Условие задачи**: решить методом Половинного деления уравнение - 0,25x3 + x - 1,2502 = 0 отрезок, содержащий корень: [0;2], точное значение: 1,0001

**Анализ задачи**: Написать код и составить блок-схему уравнения - 0,25x3 + x - 1,2502 = 0, на C++ и решить методом половинного деления

**Метод половинного деления:**

Известен интервал изоляции корня на [a, b]

Формула выражена в алгебраическом виде

Решение: а = 0; b = 2; Е = 1\*10^(-6)

Проверяем условие f(a) \* f(b) < 0, где f(x) =0,25x3 + x - 1,2502

Если условие не выполняется, выходим из метода, так как на данном отрезке нет корней.

Иначе, выполняем итерации метода:

Вычисляем среднюю точку отрезка: x0 = (a + b) / 2.

Проверяем условие f(x0) = 0 или |b - a| <= Е

Если условие выполняется, выводим значение x0 как приближенный корень.

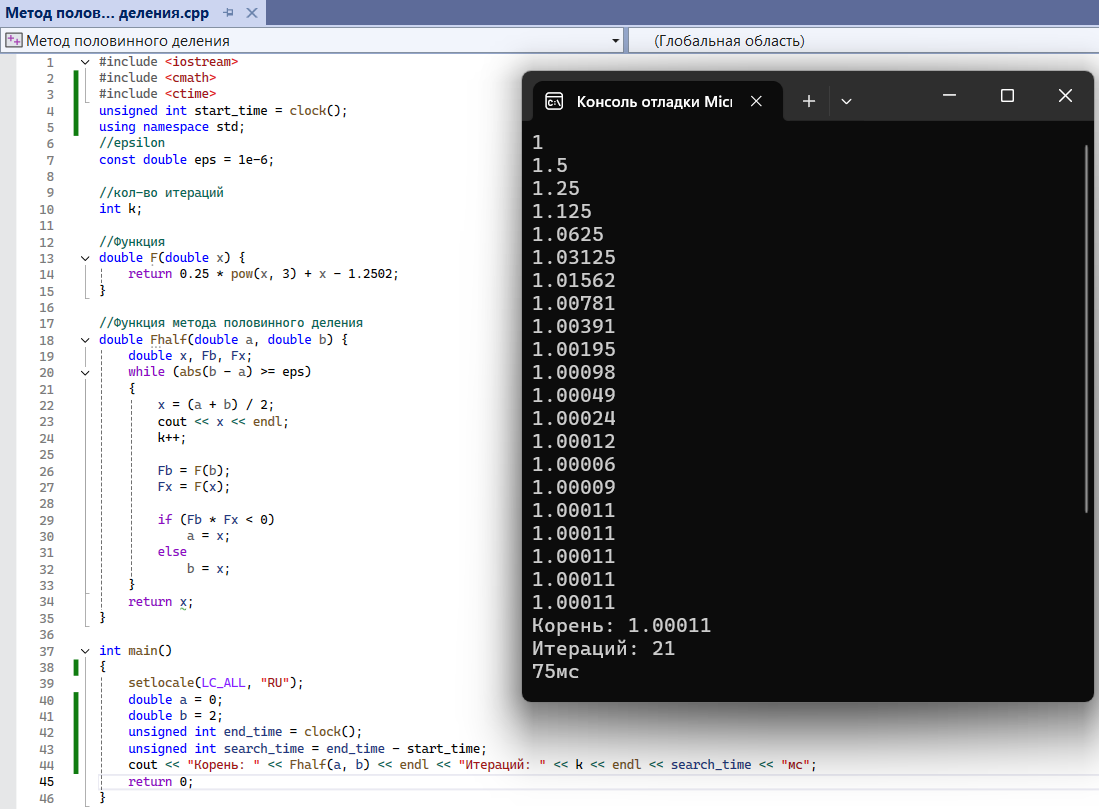
Иначе, проверяем знак f(a) \* f(c):

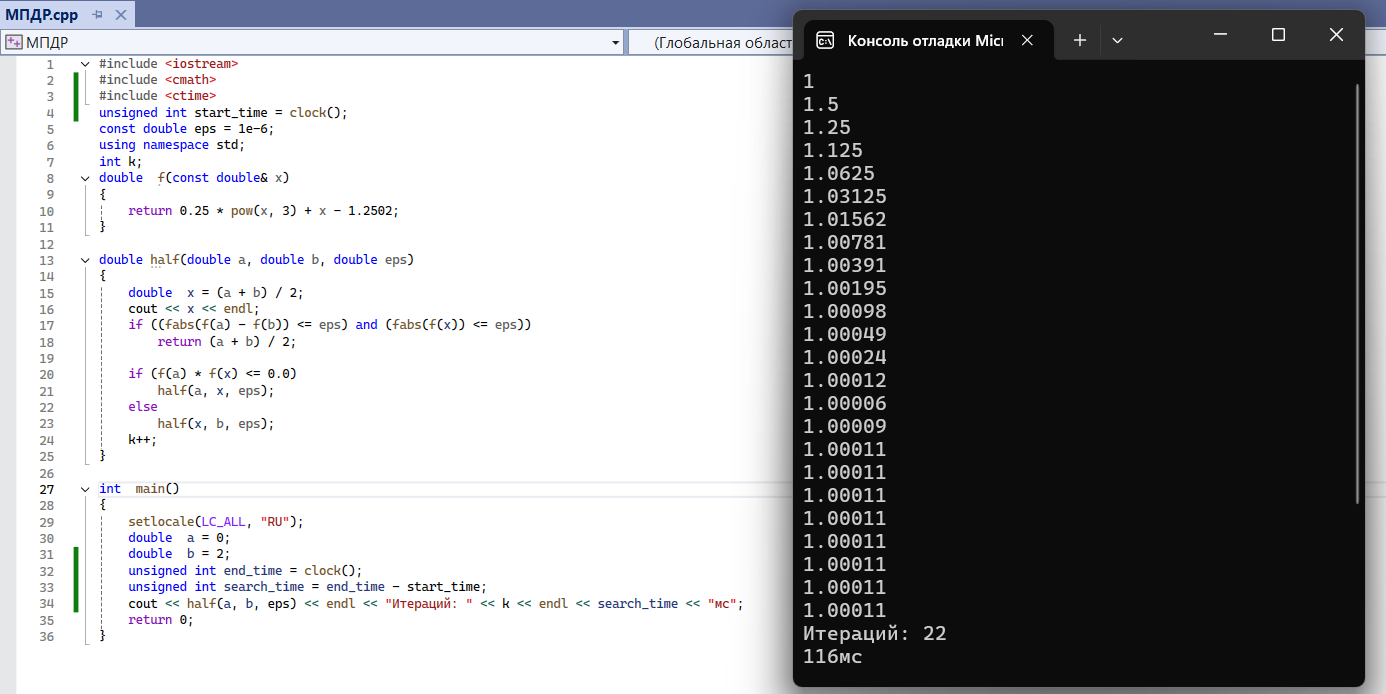
Если f(a) \* f(x0) < 0, обновляем правую границу b = x0.

Иначе, обновляем левую границу a = x0.

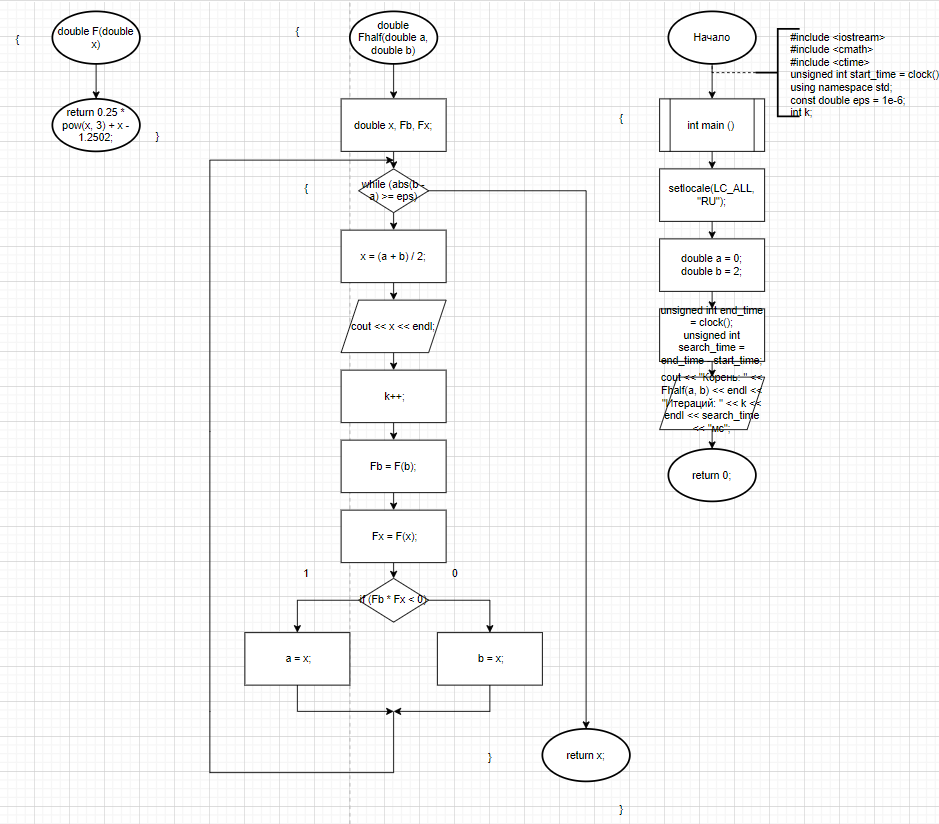
Возвращаемся к шагу 4 и продолжаем итерации до достижения заданной точности или нахождения корня.

Код C++:





Блок-схема:



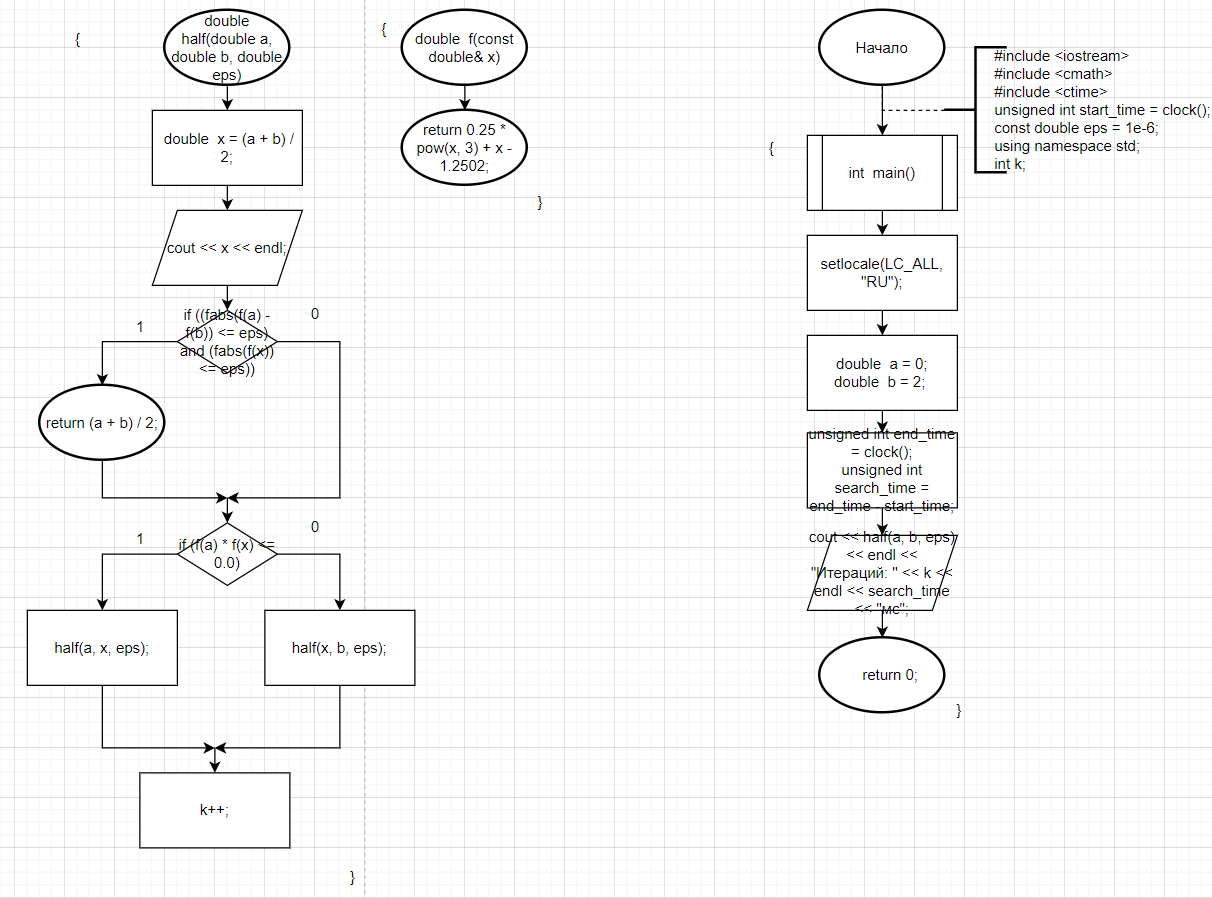
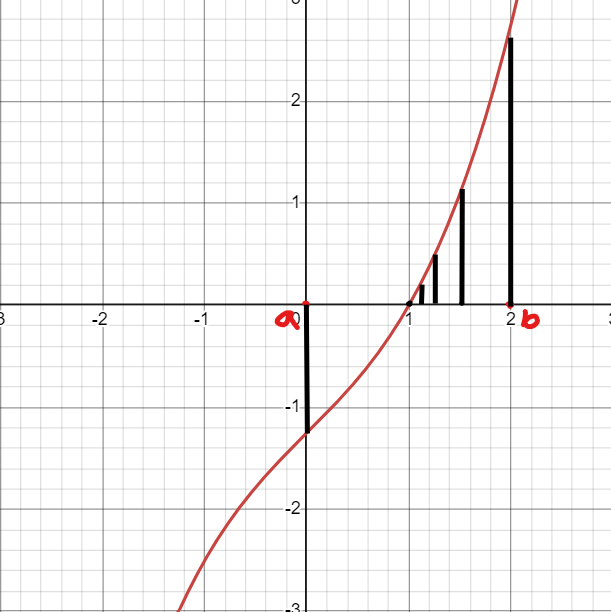


График:



Метод Итераций

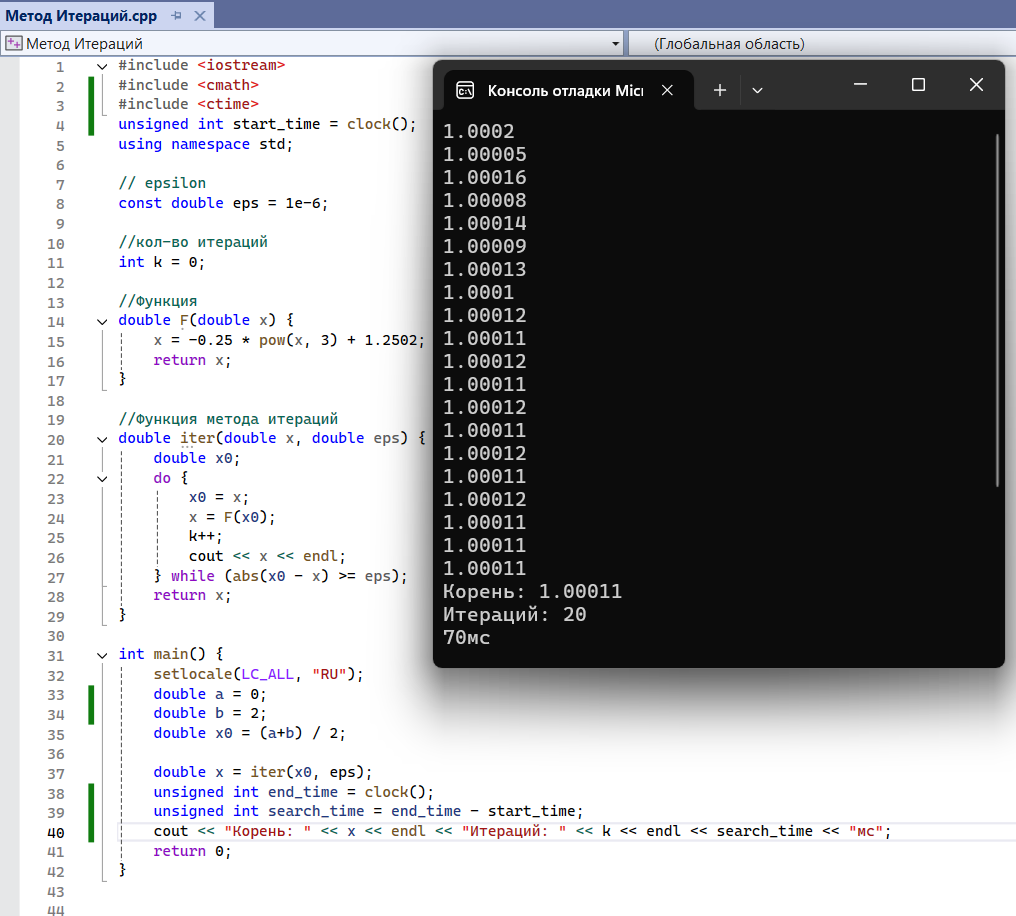
**Условие задачи**: решить методом Итераций уравнение - 0,25x3 + x - 1,2502 = 0 отрезок, содержащий корень: [0;2], точное значение: 1,0001

**Анализ задачи**: Написать код и составить блок-схему уравнения - 0,25x3 + x - 1,2502 = 0, на C++ и решить методом Итераций

**Метод итераций:**

1. Представляем исходное уравнение f(x)=0 в виде x=µ(x)
2. На отрезке [a; b] выбираем любую точку
3. Следующее приближенное значение вычисляется по формуле xi=µ(xi-1)
4. 3 шаг повторять, пока |xi-xi-1| >
5. Если |xi-xi-1| <= цикл заканчивается, xi – приближенный корень

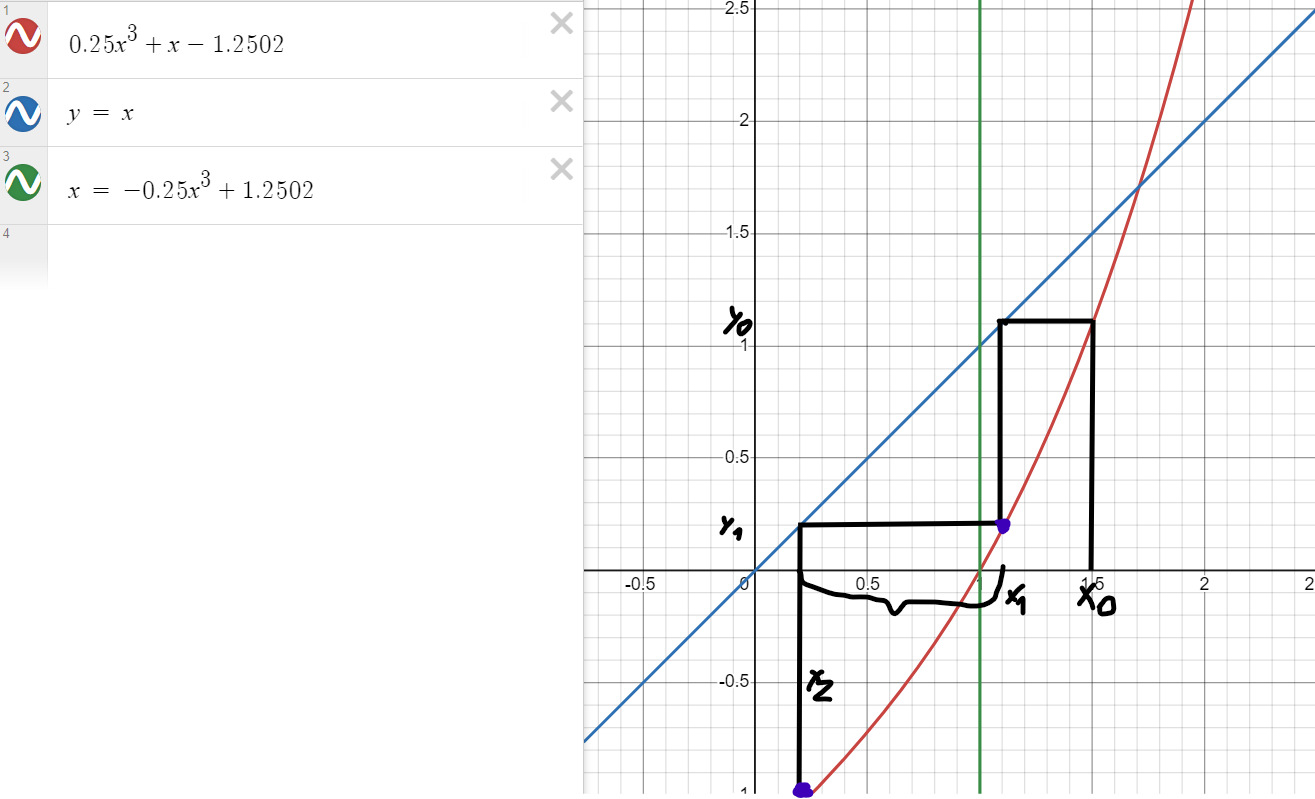
Код C++:



Блок-схема:



График:



Вывод:

В ходе выполнения лабораторной работы были изучены и применены три численных метода для решения нелинейных уравнений: метод Ньютона, метод простых итераций и метод половинного деления. Каждый из этих методов имеет свои особенности, преимущества и недостатки, которые были продемонстрированы на практике.

1. Метод Ньютона:
   * Время выполнения 66мс.
   * Преимущества: Метод Ньютона демонстрирует высокую скорость сходимости. Это делает его эффективным для задач, требующих высокой точности.
   * Недостатки: Метод требует вычисления производной функции, что может быть сложным или невозможным для некоторых функций.
2. Метод половинного деления:
   * Время выполнения 75мс, рекурсивный 116мс.
   * Преимущества: Метод половинного деления гарантирует сходимость, если начальный интервал выбран правильно. Он не требует вычисления производной и является надежным для функций, которые непрерывны на заданном интервале.
   * Недостатки: Сходимость метода половинного деления линейная, что делает его медленным по сравнению с методом Ньютона.
3. Метод простых итераций:
   * Время выполнения 70мс.
   * Преимущества: Этот метод прост в реализации и не требует вычисления производной. Он может быть полезен для функций, для которых вычисление производной затруднительно.
   * Недостатки: Сходимость метода простых итераций может быть медленной, особенно если функция имеет сложную структуру.

Github - https://github.com/DaunichOlux/University