ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ТЕЛЕКОММУНИКАЦИЙ И ИНФОРМАТИКИ»

Кафедра ПМиК

Лабораторная работа №3 по дисциплине
«Вычислительная математика» по теме:
«Метод биссекций, метод хорд и метод Ньютона»

Выполнил:
ст. гр. ИВ-823
Шиндель Э. Д.

Проверила:
Ассистент
Кафедры ПМиК
Петухова Я. В.

Новосибирск, 2020

Содержание:

1. Постановка задачи

$$f(x) = cos(x) + 2 \cdot sin(2 \cdot x) - \frac{1}{x} = 0 \qquad [0{,}5; 2]$$

Проверка на наличие корней:
f(a) · f(b) < 0 (есть хотя бы один корень)
f(a) = f(0,5) = 0,5605
f(b) = f(2) = -2,4298
f(a) · f(b) = -1,3619 < 0 → Есть хотя бы 1 корень

2. Метод биссекций

a = 0,5 b = 2 e^4 = (0,0001) или 3 итерации

$$x_i = \frac{a+b}{2}$$

Проверка: $|f(x)| < e$
Для перехода на следующую итерацию находим интервал, где происходит смена знака:
if (f(a) · f(x) < 0) a = a, b = x; [a; x]
else a = x, b = b; [x, b]

1-ая итерация:
a = 0,5 b = 2

$$x_0 = \frac{a+b}{2} = \frac{0{,}5+2}{2} = 1{,}25$$

Проверка: $|f(x_0)| = |f(1{,}25)| = 0{,}7123 >$ e $(-)$
f(a) · f(x) = f(0,5) · f(1,25) = 0,5605 · 0,7123 = 0,3992 > 0 → a = x

2-ая итерация:
a = 1,25 b = 2

$$x_1 = \frac{a+b}{2} = \frac{1{,}25+2}{2} = 1{,}625$$

Проверка: $|f(x_0)| = |f(1{,}625)| = 0{,}8860 >$ e $(-)$
f(a) · f(x) = f(1,25) · f(1,625) = 0,7123 · (-0,8860) = -0,6311 < 0 → b = x

3-я итерация:

a = 1,25   b = 1,625

$$x_2 = \frac{a+b}{2} = \frac{1{,}25 + 1{,}625}{2} = 1{,}4375$$

Проверка: $|f(x_0)| = |f(1{,}4375)| = 0{,}0359\ >\ \mathrm{e}\ (-)$

Так как мы выполнили 3 итерации, и не достигли нужной точности, то запишем ответ, сократив точность

Ответ: x = 1,4 с точностью 0,1.

3. Метод хорд

a = 0,5   b = 2 e^4 = (0,0001) или 3 итерации

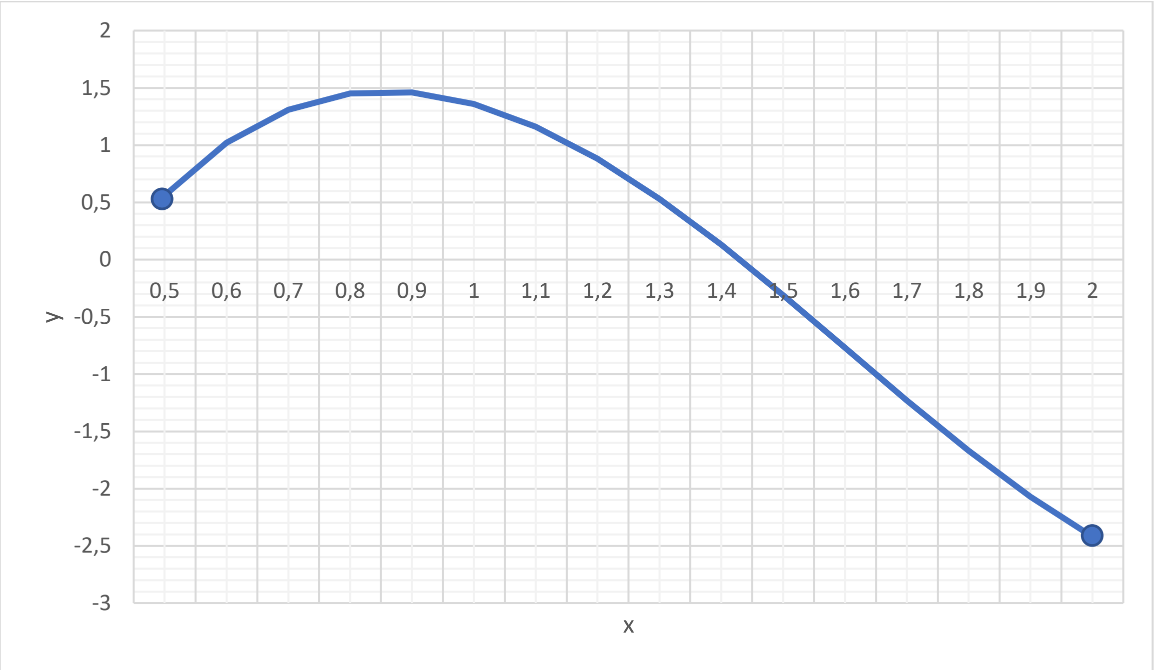
$$x_i = \frac{a \cdot f(b) - b \cdot f(a)}{f(b) - f(a)}$$

Проверка: $|x_k - x_{k-1}| < e$

Для перехода на следующую итерацию находим интервал, где происходит смена знака:

if (f(a) · f(x) < 0) a = a, b = x; [a; x]

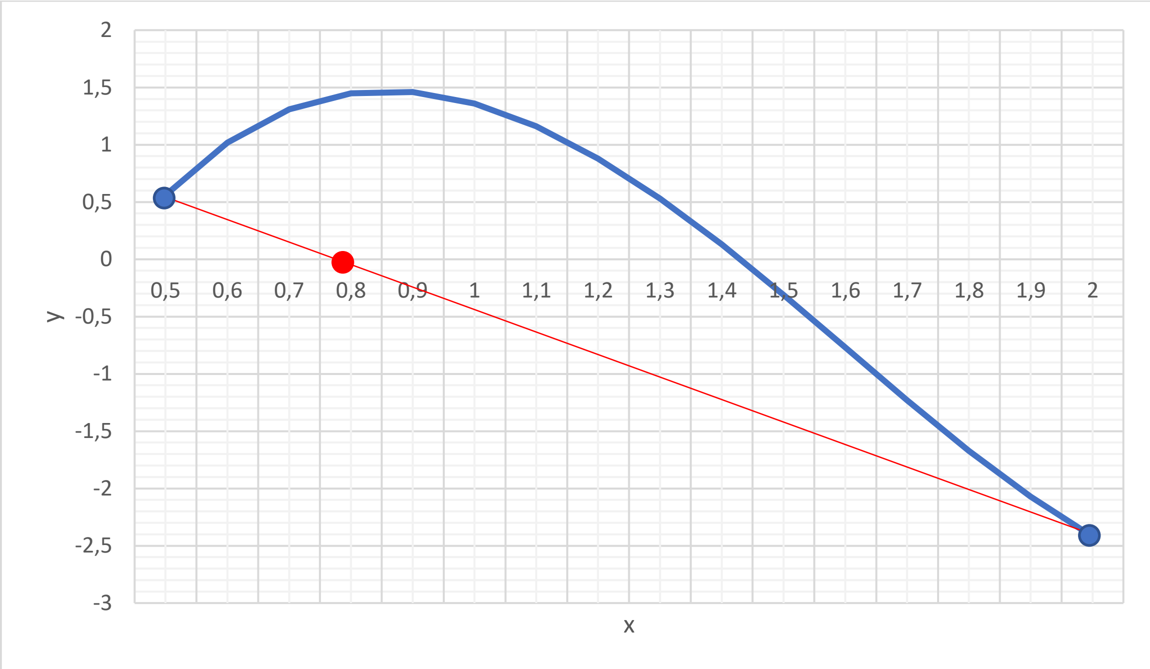
else a = x, b = b; [x, b]



1-ая итерация:

a = 0,5 b = 2

$$x_0 = \frac{a \cdot f(b) - b \cdot f(a)}{f(b) - f(a)} = \frac{0{,}5 \cdot f(2) - 2 \cdot f(0{,}5)}{f(2) - f(0{,}5)} = 0{,}7812$$

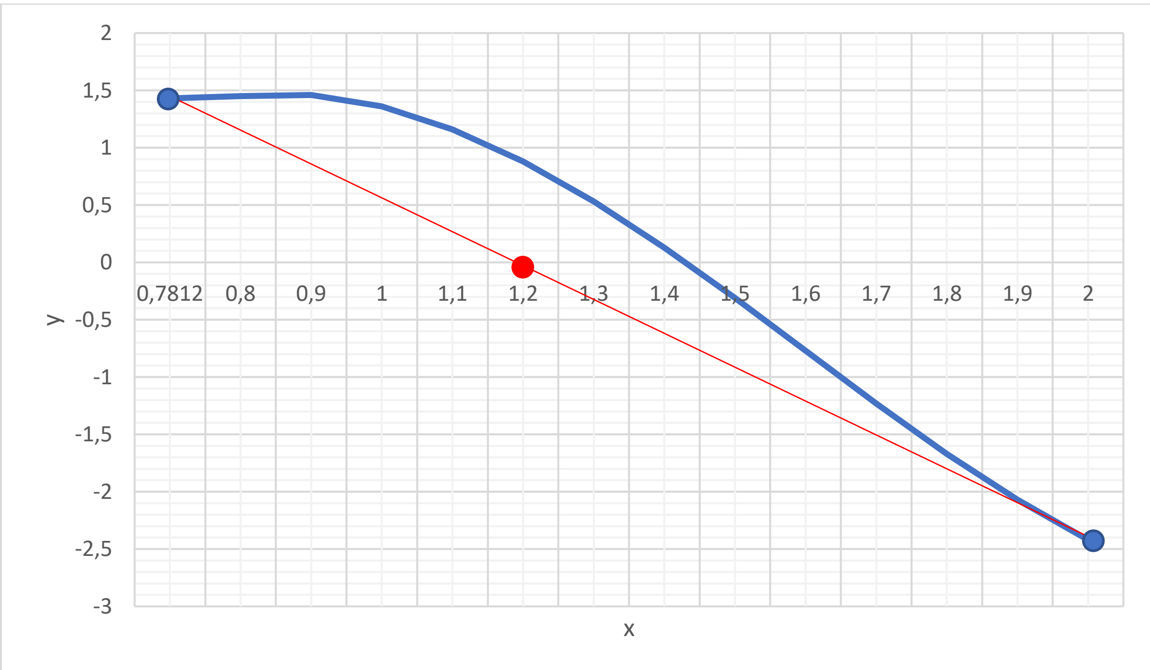


f(a) · f(x) = f(0,5) · f(0,7812) = 0,5605 · 1,4299 = 0,8402 > 0 ⟶ a = x

2-ая итерация:

a = 0,7812 b = 2

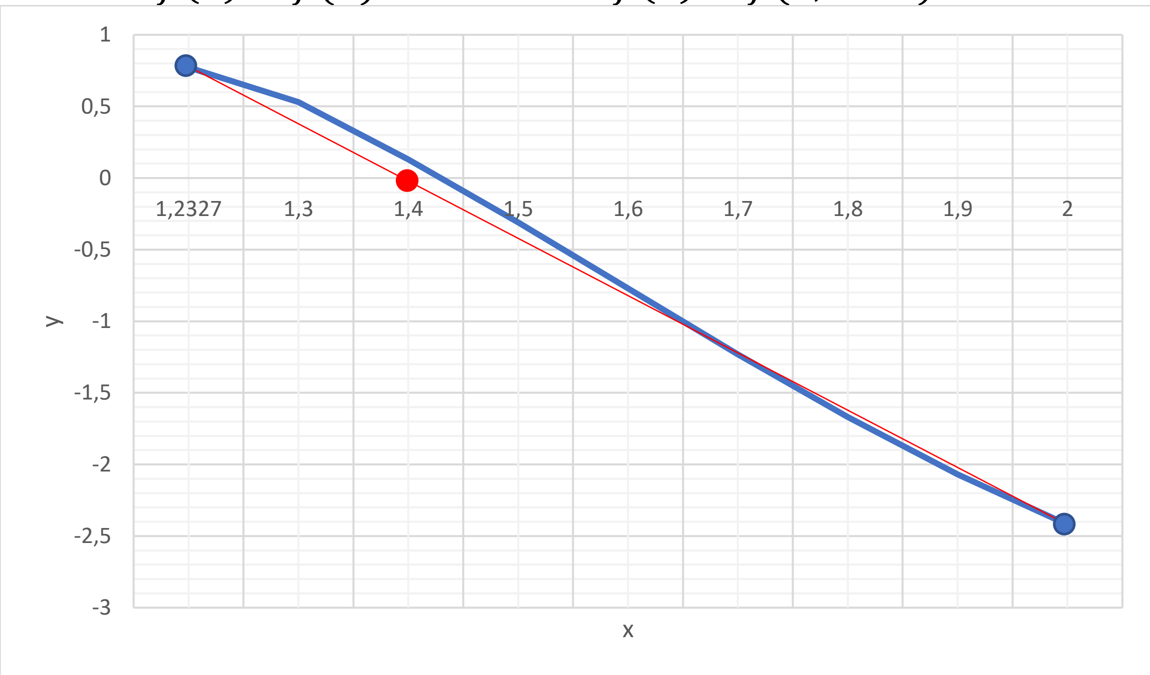
$$x_1 = \frac{a \cdot f(b) - b \cdot f(a)}{f(b) - f(a)} = \frac{0{,}7812 \cdot f(2) - 2 \cdot f(0{,}7812)}{f(2) - f(0{,}7812)} = 1{,}2327$$



Проверка: $|x_k - x_{k-1}| = |1{,}2327 - 0{,}7812| = 0{,}4515 > e\ (-)$
f(a) · f(x) = f(0,7812) · f(1,2327) = 1,4299 · 0,7721 = 1,1040 > 0 →a = x

3-я итерация:
a = 1,2327 b = 2

$$x_2 = \frac{a \cdot f(b) - b \cdot f(a)}{f(b) - f(a)} = \frac{1{,}2327 \cdot f(2) - 2 \cdot f(1{,}2327)}{f(2) - f(1{,}2327)} = 1{,}4177$$



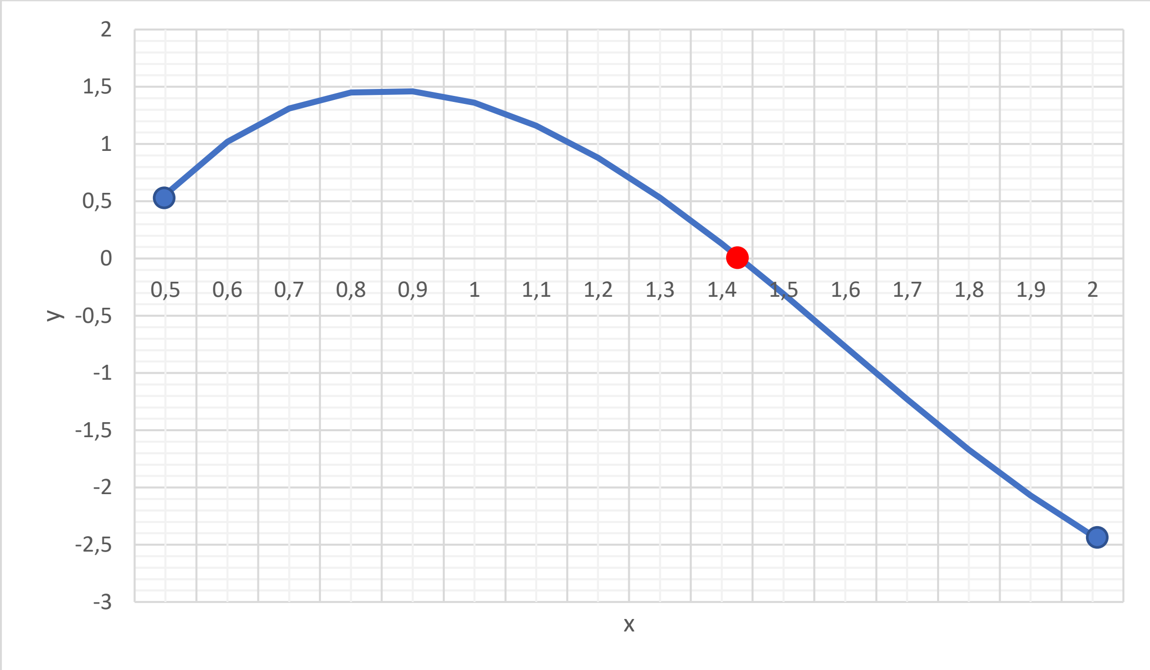
Проверка: $|x_k - x_{k-1}| = |1{,}4177 - 1{,}2327| = 0{,}185 > e\ (-)$

Так как мы выполнили 3 итерации, и не достигли нужной точности, то запишем ответ, сократив точность

Ответ: x = 1,4 с точностью 0,1.

4. Метод Ньютона

e^4 = (0,0001) или 3 итерации
Построим график и отметим точку решения.



Теперь выберем начальное приближение x_0. Обычно это один из концов отрезка. Начальное приближение должно удовлетворять условию: $f(x_0) \cdot f''(x_0) > 0$.

Найдём 1 и 2 производную функции:

$$f'(x) = -\,sin(x) + 4 \cdot cos(2 \cdot x) + \frac{1}{x^2}$$

$$f''(x) = -8\,sin(2 \cdot x) - cos(x) - \frac{2}{x^3}$$

Проверяем левый конец отрезка:

$f(0{,}5) \cdot f''(0{,}5) = \; 0{,}5605 \cdot (-23{,}6094) = -13{,}2330 < 0\ (-)$

Проверяем правый конец:

$f(2) \cdot f''(2) = \; -2{,}4298 \cdot (-6{,}2206) = 15{,}1148 > 0\ (+)$

В качестве начального приближения выбираем $x_0 = 2$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Проверка: $\left|\frac{f(xn)}{f'(x_n)}\right| < e$

1-ая итерация:

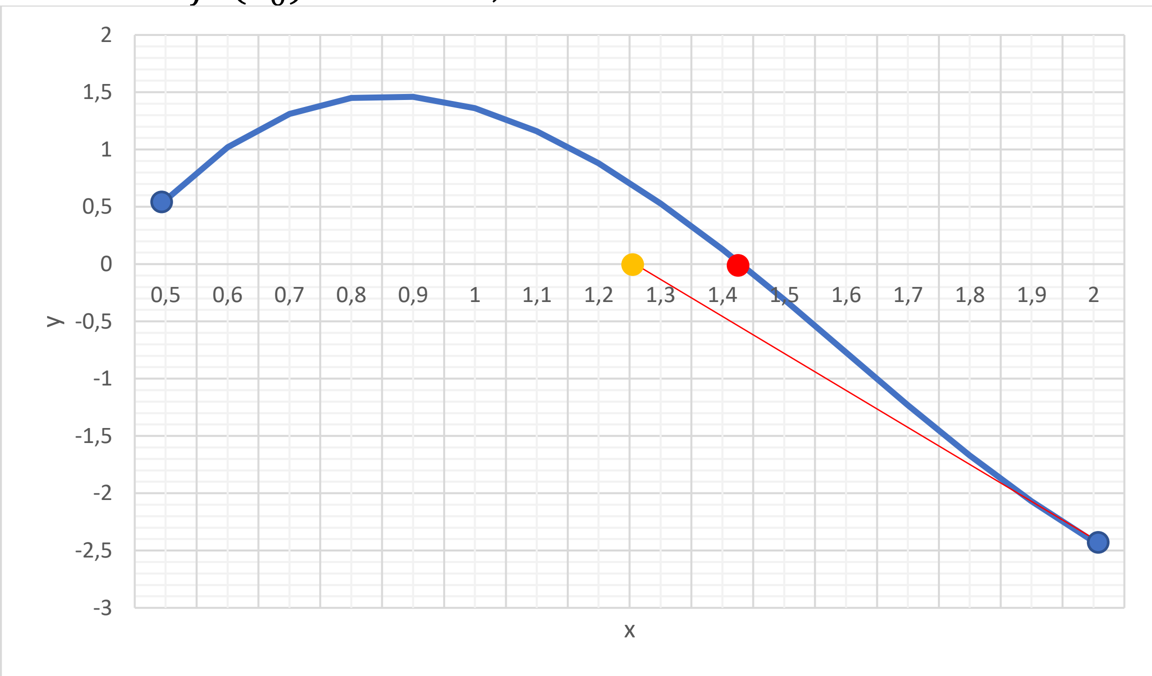
$f(x_0) = f(2) = \; -2{,}4298$

$f'(x_0) = f'(2) = \ -3{,}2739$

Проверка: $\left|\frac{f(x_0)}{f'(x_0)}\right| = \left|\frac{-2{,}4298}{-3{,}2739}\right| = 0{,}7422 > e\ (-)$

2-ая итерация

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = 2 - \frac{-2{,}4298}{-3{,}2739} = 1{,}2578$$

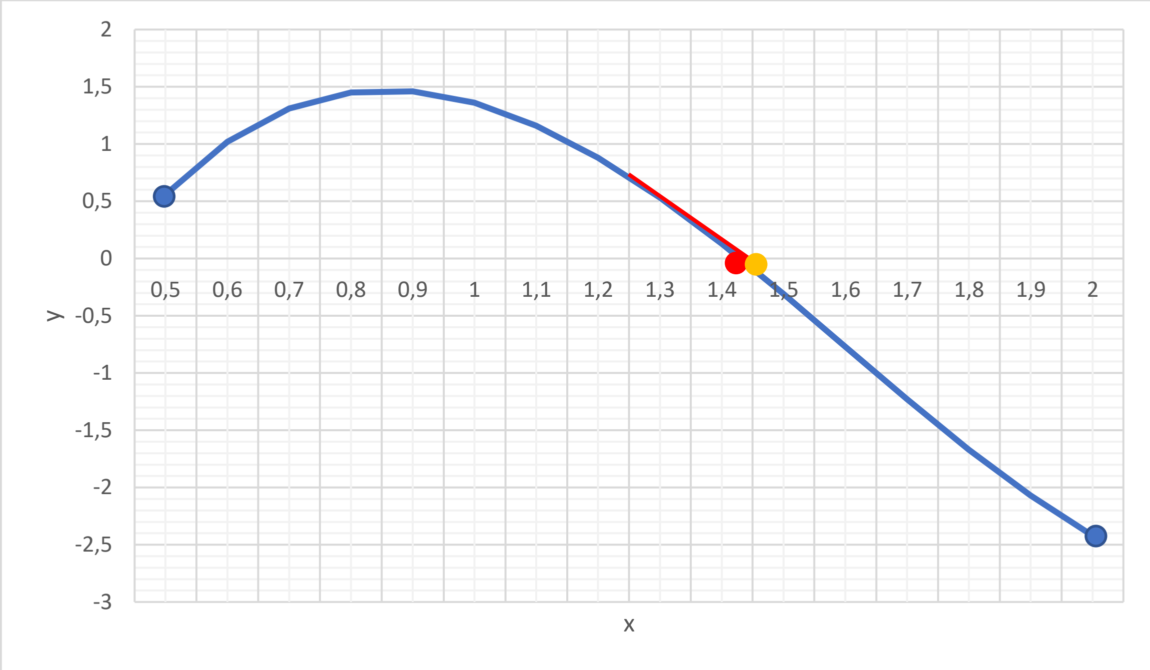


$f(x_1) = f(1{,}2578) = \ 0{,}6847$
$f'(x_1) = f'(1{,}2578) = \ -3{,}5609$

Проверка: $\left|\frac{f(x_1)}{f'(x_1)}\right| = \left|\frac{0{,}6847}{-3{,}5609}\right| = 0{,}1923 > e\ (-)$

3-я итерация:

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} = 1{,}2578 - \frac{0{,}6847}{-3{,}5609} = 1{,}4501$$



$f(x_2) = f(1{,}4501) = \ 1{,}4162$
$f'(x_2) = f'(1{,}4501) = \ -4{,}4012$

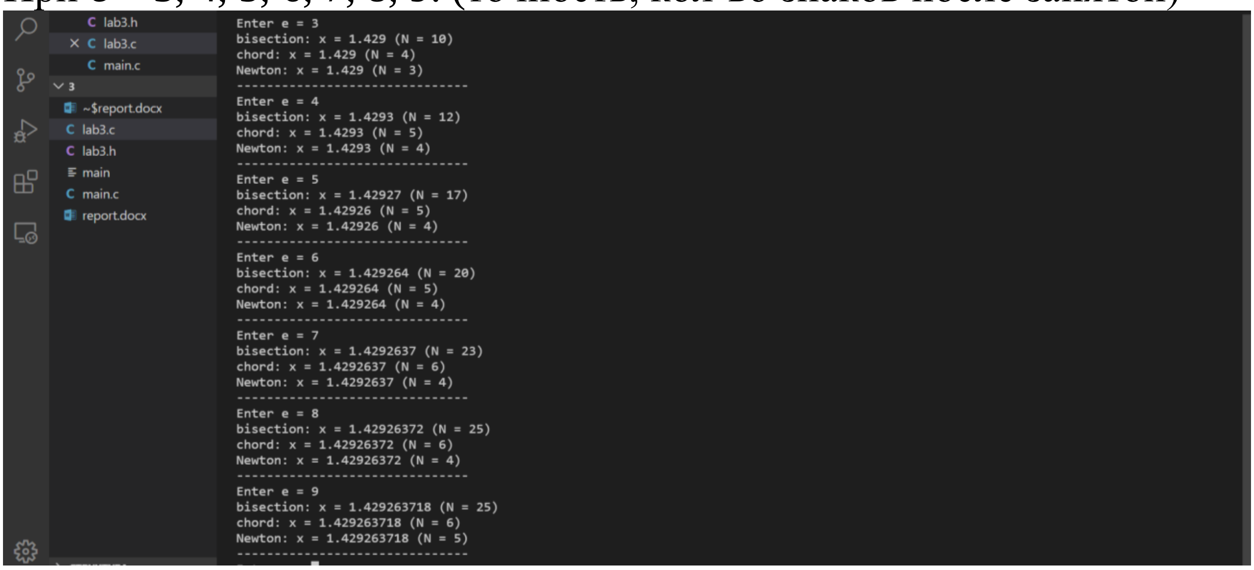
Проверка: $\left|\frac{f(x_2)}{f'(x_2)}\right| = \left|\frac{1{,}4162}{-4{,}4012}\right| = 0{,}3218 > e\ (-)$

Так как мы выполнили 3 итерации, и не достигли нужной точности, то запишем ответ, сократив точность

Ответ: x = 1,4 с точностью 0,1.

5. Пример работы

При e = 3, 4, 5, 6, 7, 8, 9. (точность, кол-во знаков после запятой)



Листинг

//lab3.c

```
#include "lab3.h"

double aim(int x) {
    double exp = 1.0;
    for (int i = 0; i < x; i++) exp /= 10;
    return exp;
}

double function(double x) {
    double f = cos(x) + 2 * sin(2 * x) - 1 / x;
    return f;
}

int check(double a, double b) {
    double f_a, f_b;
    f_a = function(a);
    f_b = function(b);
    return (((f_a * f_b) < 0) ? 0 : 1);
}

void all_in(int e) {
    double a = 0.5, b = 2.0;
    double dif = aim(e);
    if (check(a, b)) {
        printf("No roots in the interval [%.1f; %.1f]\n", a, b);
        return;
    }
    bisection(e, dif, a, b);
    chord(e, dif, a, b);
    Newton(e, dif, a, b);
}

void bisection(int e, double dif, double a, double b) {
    double x, f_a, f_x;
    int N = 0;

    do {
        x = (a + b) / 2;
        f_a = function(a);
        f_x = function(x);
        if ((f_a * f_x) < 0) b = x;
        else a = x;
        N++;
    } while (fabs(f_x) > dif);
    printf("bisection: x = %.*f (N = %d)\n", e, x, N - 1);
}

void chord(int e, double dif, double a, double b) {
    double x0 = 0.0, x1 = 0.0;
    int N = 0;

    do {
        x0 = x1;
        x1 = (a * function(b) - b * function(a)) / (function(b) - function(a));
        if ((function(a) * function(x1)) < 0) b = x1;
```

```
        else a = x1;
        N++;
    } while (fabs(x1 - x0) > dif);
    printf("chord: x = %.*f (N = %d)\n", e, x1, N - 1);
}

void Newton(int e, double dif, double a, double b) {
    double x0 = 0.0, x1 = 0.0, f1, f2;
    int N = 0;

    f2 = (function(a + dif) - 2 * function(a) + function(a - dif)) / (dif * dif);
    if ((function(a) * f2) > 0) x1 = a;
    else x1 = b;

    do {
        x0 = x1;
        f1 = (function(x0 + dif) - function(x0 - dif)) / (2 * dif);
        x1 = x0 - (function(x0) / f1);
        N++;
    } while (fabs(function(x0) / f1) > dif);
    printf("Newton: x = %.*f (N = %d)\n", e, x0, N - 1);
}
```