Группа 4310

Нигамадянов Ф.М.

Маннанов Д.И.

# ЛАБОРАТОРНАЯ РАБОТА №1-2.

# ИТЕРАЦИОННЫЕ МЕТОДЫ

# РЕШЕНИЯ НЕЛИНЕЙНЫХ УРАВНЕНИЙ.

Цель работы: научиться решать нелинейные уравнения методом простых итераций, методом Ньютона и модифицированным методом Ньютона с помощью ЭВМ.

Содержание работы:

1. Изучить метод простых итераций, метод Ньютона и модифицированный метод Ньютона для решения нелинейных уравнений.

2. На конкретном примере усвоить порядок решения нелинейных уравнений с помощью ЭВМ указанными методами.

3. Составить программу (программы) на любом языке программирования и с ее помощью решить уравнение с точностью и . Сделать вывод о скорости сходимости всех трех методов.



4. Изменить и снова решить задачу. Сделать выводы о: скорости сходимости рассматриваемых методов; влиянии точности на скорость сходимости; влиянии выбора начального приближения в методе простых итераций на скорость сходимости.



5. Составить отчет о проделанной работе.

Задание (вариант №6).

1. Доказать графическим и аналитическим методами существование одного корня нелинейного уравнения

(1)

на отрезке .

2. Построить рабочие формулы метода простых итераций, метода Ньютона и модифицированного метода Ньютона, реализующие процесс поиска корня нелинейного уравнения (1) на указанном отрезке.

3. Составить программу (программы) на любом языке программирования, реализующие построенные итерационные процессы.

Решение.

1. Докажем графическим методом единственность корня нелинейного уравнения (1) на заданном отрезке. Из графика функции = на Рис.1 видно, что функция пересекает ось OX в точке, являющейся приближенным значением корня нелинейного уравнения (1). Но так как данная функция имеет сложный аналитический вид, то преобразуем уравнение (1) к виду и построим два графика и , имеющих более простой аналитический вид (Рис.2). Абсцисса точки пересечения графиков является приближенным значением корня. Заметим, что графический метод показывает количество корней исходного уравнения, но не доказывает единственность корня на отрезке.

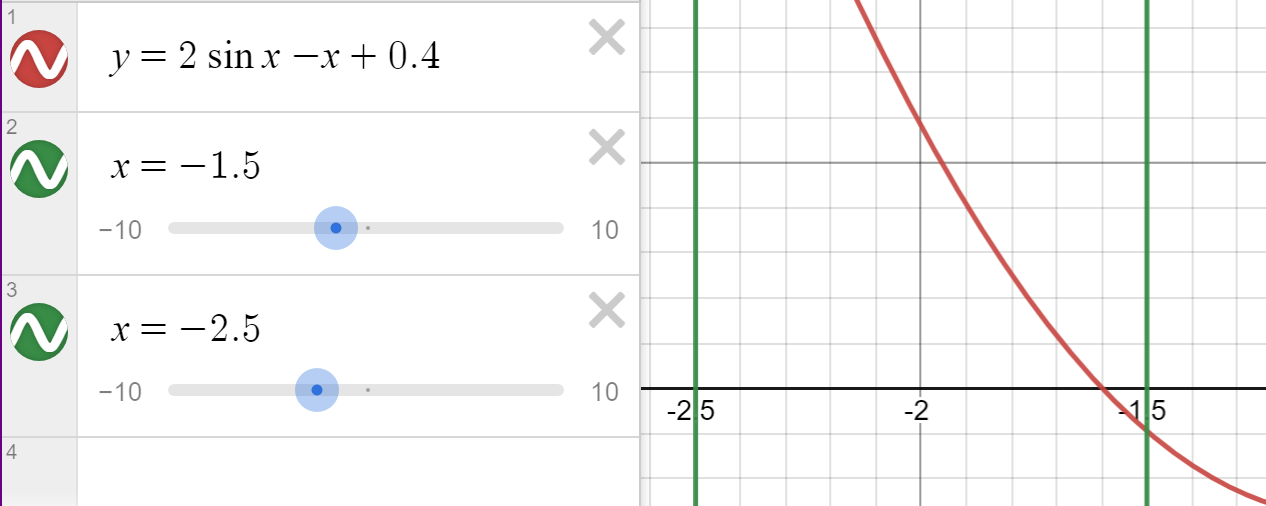


Рис. 1

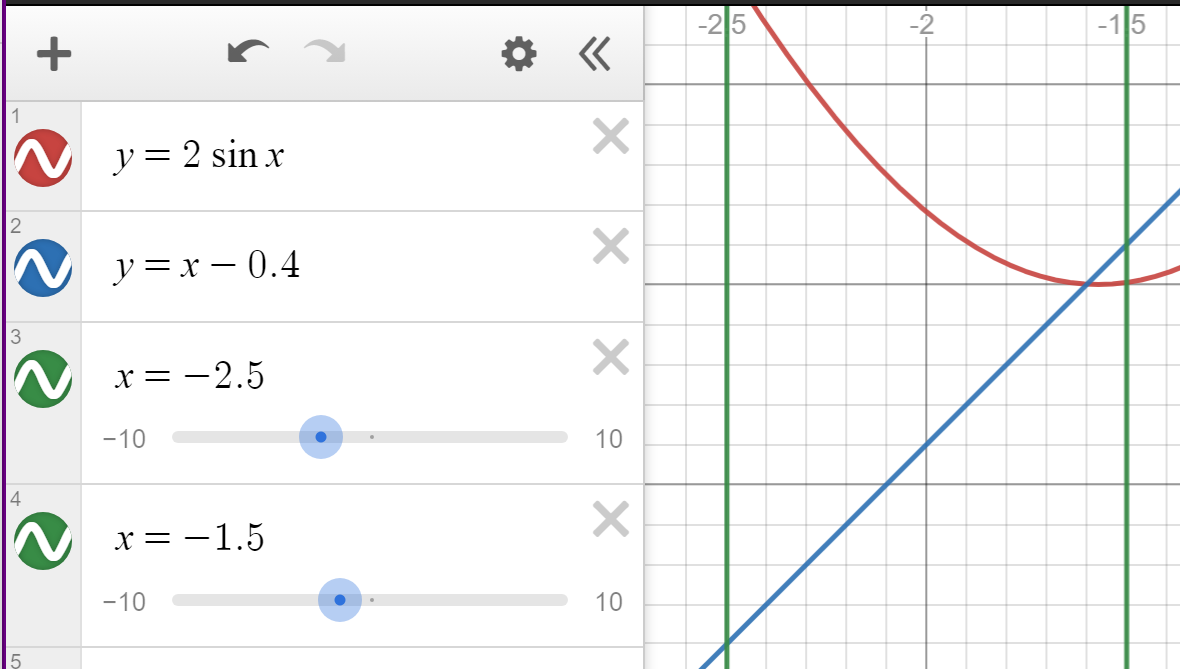


Рис. 2

Аналитический метод. Функция непрерывна на отрезке , имеет на концах отрезка разные знаки (), а производная функции не меняет знак на отрезке (). Следовательно, нелинейное уравнение (1) имеет на указанном отрезке единственный корень.

2. Метод простых итераций.

Построим функцию . Константа выбирается из достаточного условия сходимости

(2)

Если производная , то значение выбирается из интервала , если производная , то – из интервала . Так как для рассматриваемого примера всюду отрицательна на отрезке , то придавая переменной различные значения из интервала и выбирая наименьший интервал , получим . Выбираем произвольное значение из этого интервала. Пусть . Тогда рабочая формула метода простых итераций будет иметь вид:

(3)

Итерационный процесс (3) можно начать, задав произвольное начальное приближение , мы выберем . Итерационный процесс (3) заканчивается при одновременном выполнении двух условий:

и . (4)

В этом случае значение является приближенным значением корня нелинейного уравнения (1) на отрезке .

Метод Ньютона. В качестве начального приближения здесь выбирается правый или левый конец отрезка, в зависимости от того, в котором выполняется достаточное условие сходимости метода Ньютона вида:



(5)



Заметим, что в точке условие (5) не выполняется, а в точке − выполняется. Следовательно, в качестве начального приближения выбирается точка . Рабочая формула метода Ньютона для данного уравнения запишется так:



(6)

Условия выхода итерационного процесса (6) аналогичны условиям (4) метода простых итераций.

Модифицированный метод Ньютона. Начальное приближение выбирается аналогично методу Ньютона, т.е. . Рабочая формула модифицированного метода Ньютона для данного примера запишется так:



(7)

Условия выхода итерационного процесса (7) аналогичны условиям (4) метода простых итераций.

Замечание: для того, чтобы сделать вывод о скорости сходимости методов, необходимо в каждом методе выбирать одинаковое начальное приближение.

Листинг программы:

#include <iostream>

#include "locale.h"

#include <math.h>

#include <iomanip>

using namespace std;

void method\_simple\_implications() // Метод простых итераций

{

int n = 0;

float x = -2.5, y, z, с = 1, mod1, mod2, eps = 0.001, d = 0.01;

do

{

z = x; // z - это Xn

y = z + с \* (2\*sin(z)-z+0.4); // y - это Xn + 1

mod1 = fabs(y - z);

mod2 = fabs(2 \* sin(y) - y + 0.4);

x = y;

cout << "|" << setw(5) << n << setw(5) << "|" << setw(12) << z << setw(5) << "|" <<

setw(12) << y << setw(5) << "|" << setw(14) << mod1 << setw(8) << "|" << setw(14) << mod2 << setw(8)

<< "|" << endl;

n += 1;

} while ((mod1 > eps) (mod2 > d));

}

void method\_Newton() // Метод Ньютона

{

int n = 0;

float x = -2.5, y, z, mod1, mod2, eps = 0.001, d = 0.01;

do

{

z = x; // z - это Xn

y = z - (2\*sin(z)-z+0.4)/(2\*cos(z)-1); // y - это Xn + 1

mod1 = fabs(y - z);

mod2 = fabs(2 \* sin(y) - y + 0.4);

x = y;

cout << "|" << setw(5) << n << setw(5) << "|" << setw(12) << z << setw(5) << "|" <<

setw(12) << y << setw(5) << "|" << setw(14) << mod1 << setw(8) << "|" << setw(14) << mod2 << setw(8)

<< "|" << endl;

n += 1;

} while ((mod1 > eps) (mod2 > d));

}

void modification\_method\_Newton() // Модифицированный Метод Ньютона

{

int n = 0;

float x = -2.5, x0=x, y, z, mod1, mod2, eps = 0.001, d = 0.01;

do

{

z = x; // z - это Xn

y = z - (2 \* sin(z) - z + 0.4) / (2 \* cos(x0) - 1); // y - это Xn + 1

mod1 = fabs(y - z);

mod2 = fabs(2 \* sin(y) - y + 0.4);

x = y;

cout << "|" << setw(5) << n << setw(5) << "|" << setw(12) << z << setw(5) << "|" <<

setw(12) << y << setw(5) << "|" << setw(14) << mod1 << setw(8) << "|" << setw(14) << mod2 << setw(8)

<< "|" << endl;

n += 1;

} while ((mod1 > eps) || (mod2 > d));

}

void show\_table()

{

cout << "-----------------------------------------------------------------------------------------\n";

cout << "| n | Xn | Xn + 1 | |(Xn+1 - Xn)| | |f(Xn + 1)| |\n";

cout << "-----------------------------------------------------------------------------------------\n";

}

void func()

{

int n;

while (true)

{

cout << "Команда 1 - поиск корня на отрезке по Методу простых итераций" << endl;

cout << "Команда 2 - поиск корня на отрезке по Методу Ньютона" << endl;

cout << "Команда 3 - поиск корня на отрезке по Модифицированному Методу Ньютона" << endl;

cout << "Введите команду: "; cin >> n;

switch (n)

{

case 1:

cout << "\nМетод простых итераций" << endl;

show\_table();

method\_simple\_implications();

cout << "\nПроцесс поиска корня на отрезке прошел успешно !" << endl;

break;

case 2:

cout << "\nМетод Ньютона" << endl;

show\_table();

method\_Newton();

cout << "\nПроцесс поиска корня на отрезке прошел успешно !" << endl;

break;

case 3:

cout << "\nМодифицированный Метод Ньютона" << endl;

show\_table();

modification\_method\_Newton();

cout << "\nПроцесс поиска корня на отрезке прошел успешно !" << endl;

break;

default:

cout << "Ошибка ввода !";

break;

}

cout << "\n";

// system("pause");

// system("cls");

}

}

int main()

{

setlocale(0, "");

cout << "Лабораторная работа по Вычислительной математике No1-2\n" << endl;

cout << "Выполнили студенты Маннанов и Нигамадянов гр. 4310\n" << endl;

cout << "Вариант No6\nНелинейное уравнение: f(x) = 2sin(x)-x+0.4=0\n Отрезок: [ -2.5, -1.5 ]\n" << endl;

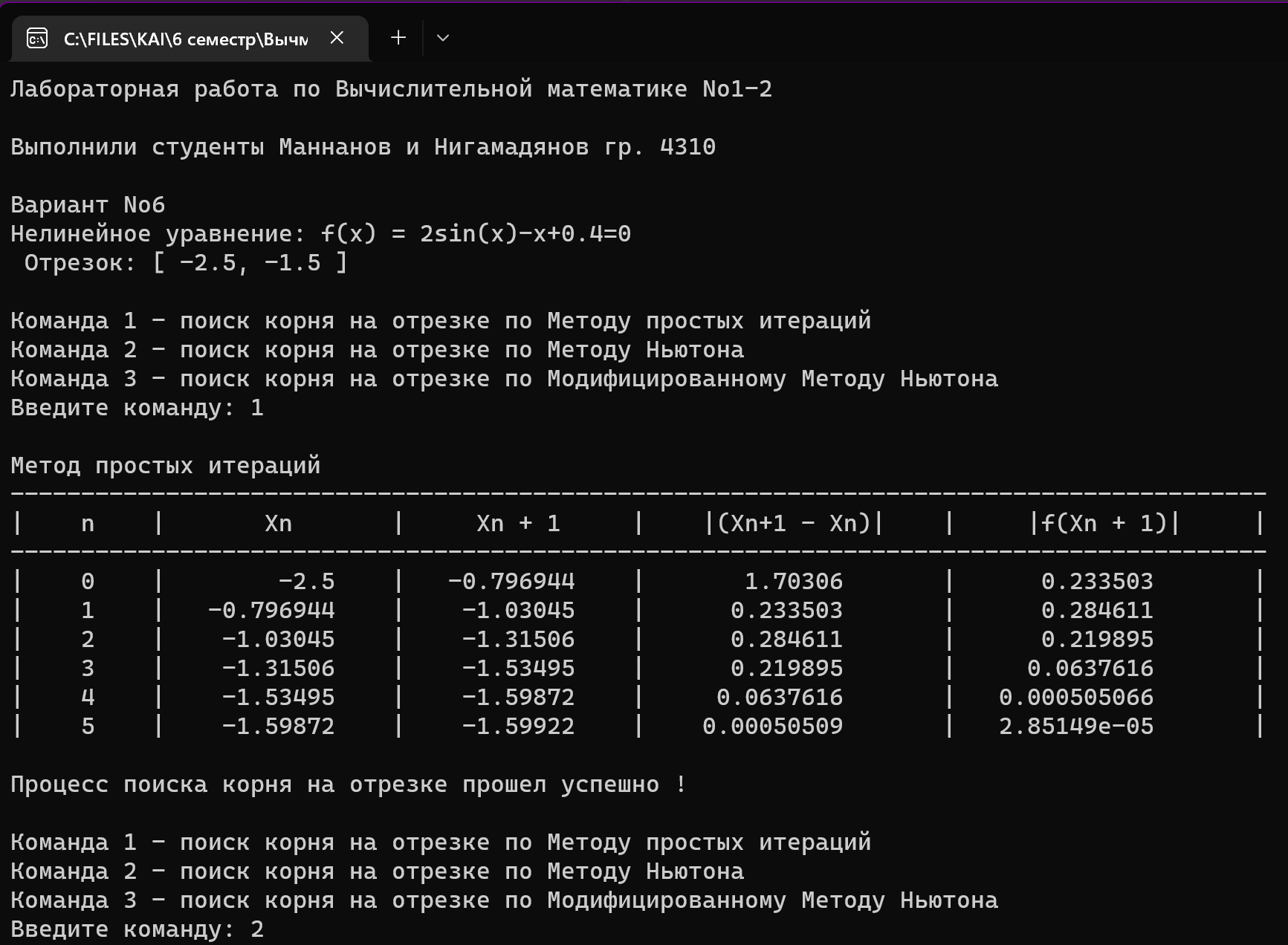
func();

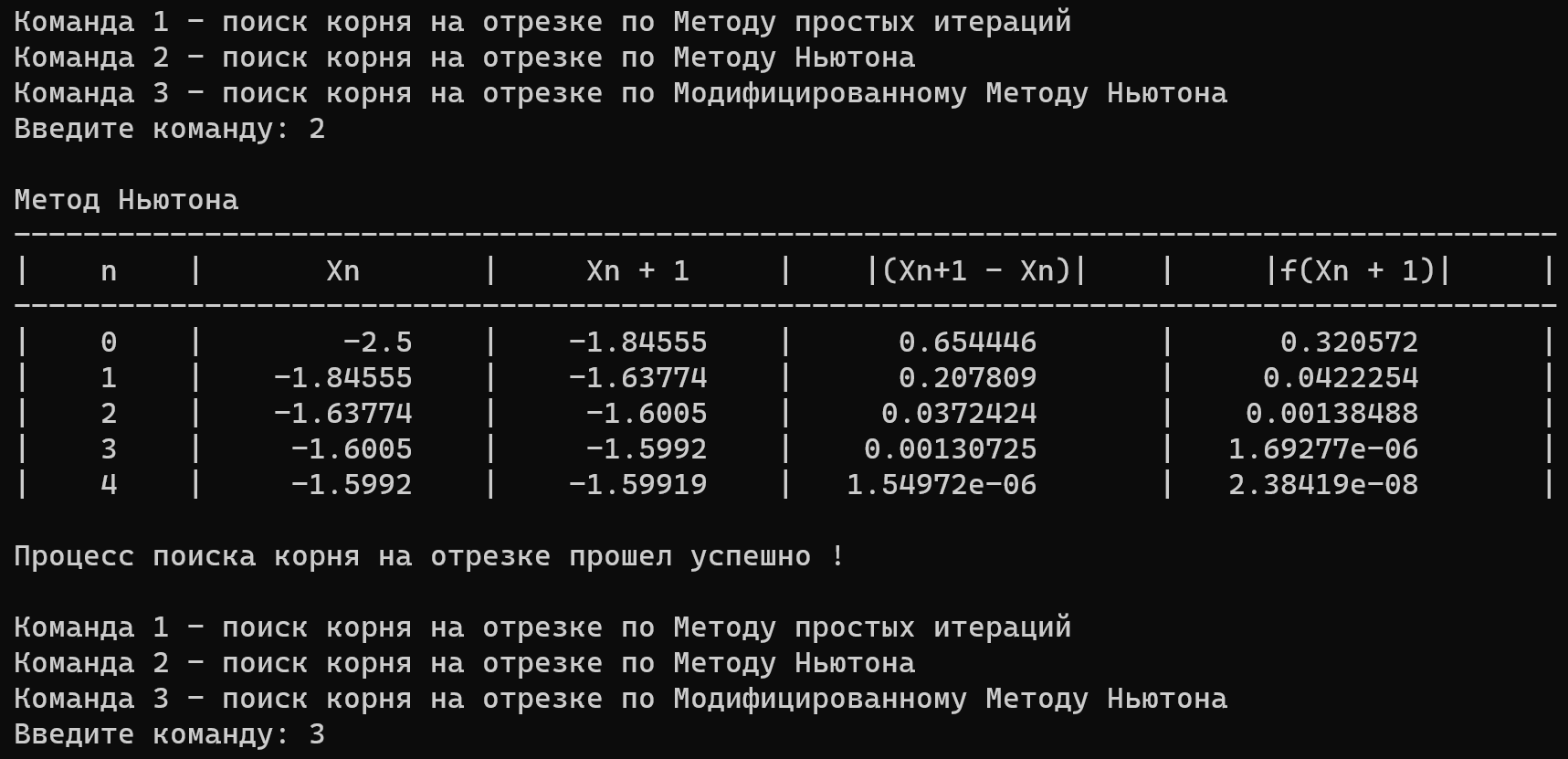
system("pause");

return 0;

}

Работа программы:





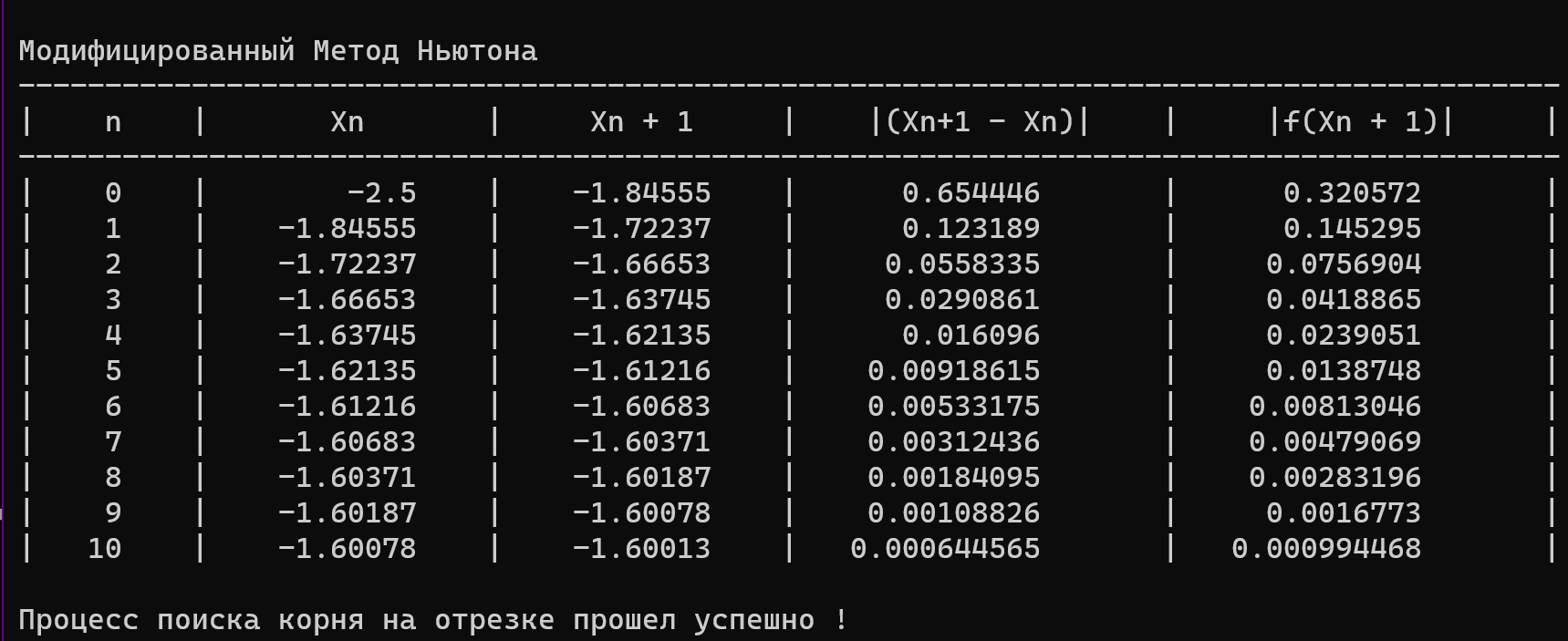


Таблица результатов вычислений по трем методам:

**Метод простых итераций**

| n |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | -2.5 | -0.796944 | 1.70306 | 0.233503 |
| 1 | -0.796944 | -1.03045 | 0.233503 | 0.284611 |
| 2 | -1.03045 | -1.31506 | 0.284611 | 0.219895 |
| 3 | -1.31506 | -1.53495 | 0.219895 | 0.0637616 |
| 4 | -1.53495 | -1.59872 | 0.0637616 | 0.000505066 |
| 5 | -1.59872 | -1.59922 | 0.00050509 | 0.0000285149 |

**Метод Ньютона**

| n |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | -2.5 | -1.84555 | 0.654446 | 0.320572 |
| 1 | -1.84555 | -1.03045 | 0.207809 | 0.0422254 |
| 2 | -1.03045 | -1.63774 | 0.0372424 | 0.00138488 |
| 3 | -1.63774 | -1.5992 | 0.00130725 | 0.00000169277 |
| 4 | -1.5992 | -1.59919 | 0.00000154972 | 0.0000000238419 |

**Модифицированный метод Ньютона**

| n |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | -2.5 | -1.84555 | 0.654446 | 0.32057 |
| 1 | -1.84555 | -1.72237 | 0.123189 | 0.145295 |
| 2 | -1.72237 | -1.66653 | 0.0558335 | 0.0756904 |
| 3 | -1.66653 | -1.63745 | 0.0290861 | 0.0418865 |
| 4 | -1.63745 | -1.62135 | 0.016096 | 0.0239051 |
| 5 | -1.62135 | -1.60683 | 0.00918615 | 0.0138748 |
| 6 | -1.60683 | -1.60371 | 0.00533175 | 0.00813046 |
| 7 | -1.60371 | -1.60187 | 0.00312436 | 0.00479069 |
| 8 | -1.60187 | -1.60187 | 0.00184095 | 0.00283196 |
| 9 | -1.60187 | -1.60078 | 0.00108826 | 0.0016773 |
| 10 | -1.60078 | -1.60013 | 0.000644565 | 0.000994468 |

Вывод: в результате решения нелинейного уравнения (1) на указанном отрезке тремя методами при начальном приближении с точностью и получены следующие результаты:



* методом простых итераций ;
* методом Ньютона ;
* модифицированным методом Ньютона .