

Московский государственный технический

университет им. Н.Э. Баумана.

Факультет «Информатика и системы управления»

Кафедра ИУ5. Курс «Основы информатики»

Отчет по лабораторной работе № 4

«Нахождение корней нелинейного уравнения»

Выполнил:

студент группы ИУ5-13

Терентьев Владислав

Подпись и дата:

Проверил:

преподаватель каф. ИУ5

Козлов А. Д.

Подпись и дата:

Москва, 2018 г.

1. Постановка задачи

Написать программу, которая должна тремя способами (простой итерацией, половинным делением и методом Ньютона) найти с определенной погрешностью корень уравнения:
 $x - \cos x = 0$

2. Разработка алгоритма

- Метод простой итерации:

Преобразование исходного уравнения к виду $x = g(x)$ и выполнение итераций $x_{n+1} = g(x)$, пока x и $g(x)$ не станут равными между собой с допустимой погрешностью. Это будет корнем исходного уравнения.

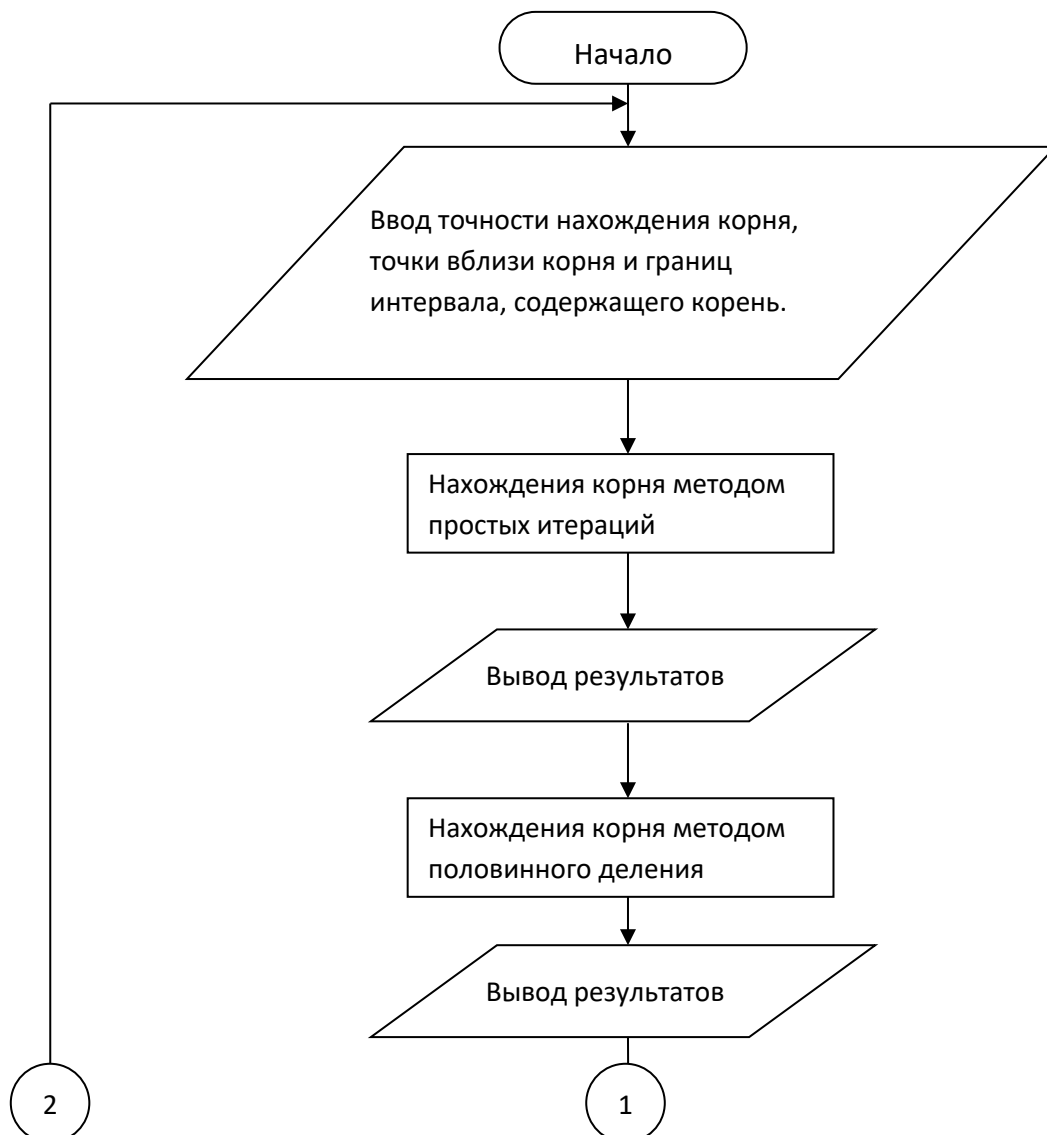
- Метод половинного деления:

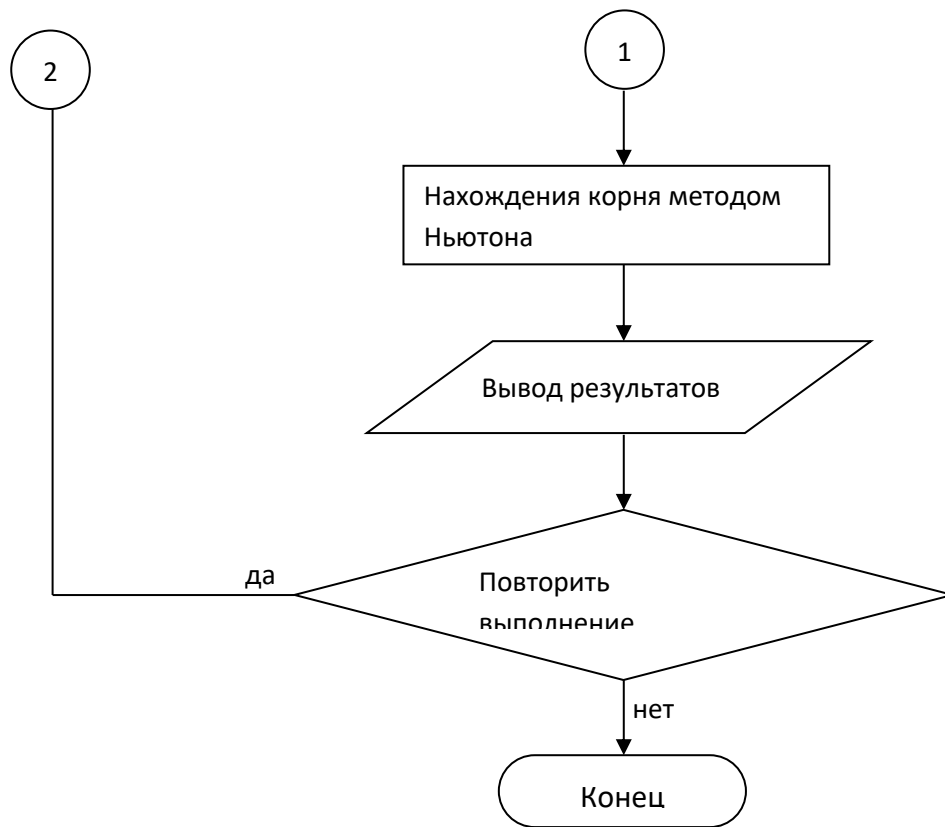
Вычисление значения исходной функции ($f(x) = x - \cos(x)$) в середине интервала и исключение той половины, которая не может содержать корень исходного уравнения (знаки на границах этого интервала одинаковы), пока не будет найден ноль функции с определенной погрешностью.

- Метод Ньютона:

Выполнение итерационной процедуры вычисления $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$, пока это не будет корнем исходного уравнения с определенной погрешностью.

Укрупненная схема алгоритма:





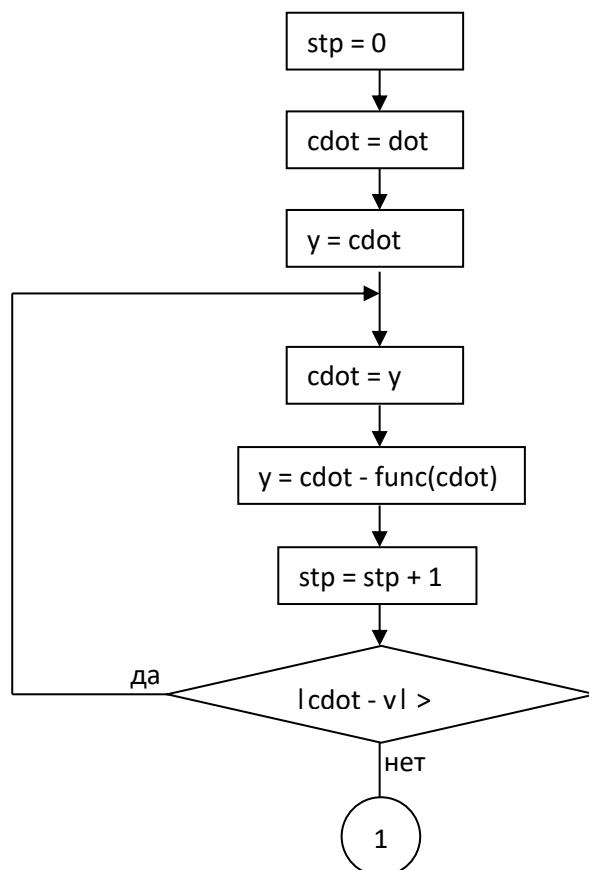
Описание переменных: переменные типа int: ex – для цикла; stp – счетчик шагов.

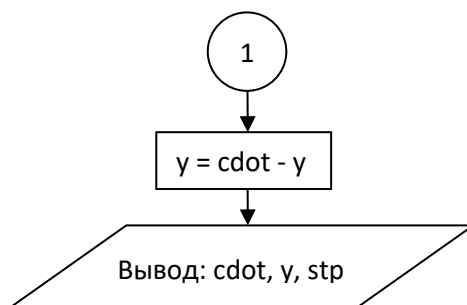
типа double: dot – точка; cdot – копия точки; aim – точность; lf – левая граница; rg – правая граница; x – аргумент; y – значение функции.

Описание функций: функция func имеет на вход параметр типа double и возвращает значение типа double выражения: $x - \cos(x)$, где x – входной параметр.

функция derive имеет на вход параметр типа double и возвращает значение типа double выражения: $\sin(x) + 1$, где x – входной параметр.

Уточненная схема блока “Нахождения корня методом простых итераций”:





3. Текст программы

```

#include "pch.h"
#include <iostream>
#include <cmath>
#include <iomanip>

using namespace std;

double func(double);
double deriv(double);

int main()
{
    setlocale(LC_ALL, "Russian");

    int ex, //переменная для цикла
        stp; //счетчик шагов

    double dot, //точка
        cdot, //копия точки
        aim, //точность
        lf, //левая граница
        rg, //правая граница
        x, //аргумент
        y; //значение функции

    do {
        cout << "Введите точность нахождения корня. Например: 1e-13" << endl;
        cin >> aim;
        cout << "Введите точку вблизи корня. Например: 0" << endl;
        cin >> dot;
        cout << "Введите границы интервала, содержащего корень. Например: 0 1" <<
endl;
        cin >> lf >> rg;

        stp = 0;
        cdot = dot;
        y = cdot;
        do {
            cdot = y;
            y = cdot - func(cdot);
            stp++;
        } while (abs(cdot - y) > aim);

        y = cdot - y;
        cout << endl << "Метод простых итераций:" << endl << "Значение функции в
точке " << setprecision(int(abs(log10(aim)) + 3)) << cdot << " равно: " << y << " Кол-во
шагов: " << stp << endl << endl;

        stp = 0;
        do {
            x = (lf + rg) / 2;
            y = func(x);
            if (y > 0) {

```

```

        rg = x;
    }
    else {
        lf = x;
    }
    stp++;
} while (abs(y) > aim);

cout << "Метод половинного деления:" << endl << "Значение функции в точке "
<< x << " равно: " << y << " Кол-во шагов: " << stp << endl << endl;

stp = 0;
cdot = dot;
x = cdot;
do {
    cdot = x;
    x = cdot - func(cdot) / deriv(cdot);
    y = func(x);
    stp++;
} while (abs(y) > aim);

cout << "Метод Ньютона:" << endl << "Значение функции в точке " << x << "
равно: " << y << " Кол-во шагов: " << stp << endl << endl << "Вы хотите продолжить или
выйти из программы? 1 - продолжить, 0 - выйти." << endl;
    cin >> ex;
} while (ex != 0);

return 0;
}

double func(double x) {
    return x - cos(x);
}

double deriv(double x) {
    return sin(x) + 1;
}

```

4. Анализ результатов

```

C:\Program Files\Microsoft Visual Studio\2019\Community\VC\Tools\MSVC\14.29.30133\bin\Hostx64-
Введите точность нахождения корня. Например: 1e-13
1e-13
Введите точку вблизи корня. Например: 0
0
Введите границы интервала, содержащий корень. Например: 0 1
-2 4

Метод простых итераций:
Значение функции в точке 0.7390851332152145 равно: 9.015010959956271e-14 Кол-во шагов: 76

Метод половинного деления:
Значение функции в точке 0.7390851332151556 равно: -8.548717289613705e-15 Кол-во шагов: 45

Метод Ньютона:
Значение функции в точке 0.7390851332151607 равно: 0 Кол-во шагов: 5

Вы хотите продолжить или выйти из программы? 1 - продолжить, 0 - выйти.
0

```