МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ПЕРМСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ЛАБОРАТОРНАЯ РАБОТА  
на тему «Разработка программ решения уравнения»

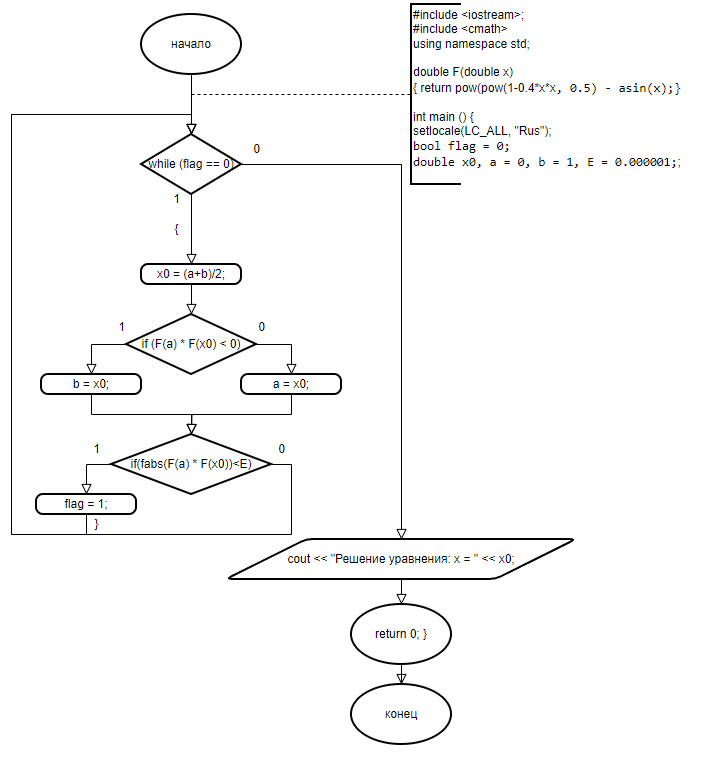
Выполнила: студентка группы РИС-23-2Б Жилина Анастасия Александровна

Проверила: доцент кафедры ИТАС О.А.Полякова

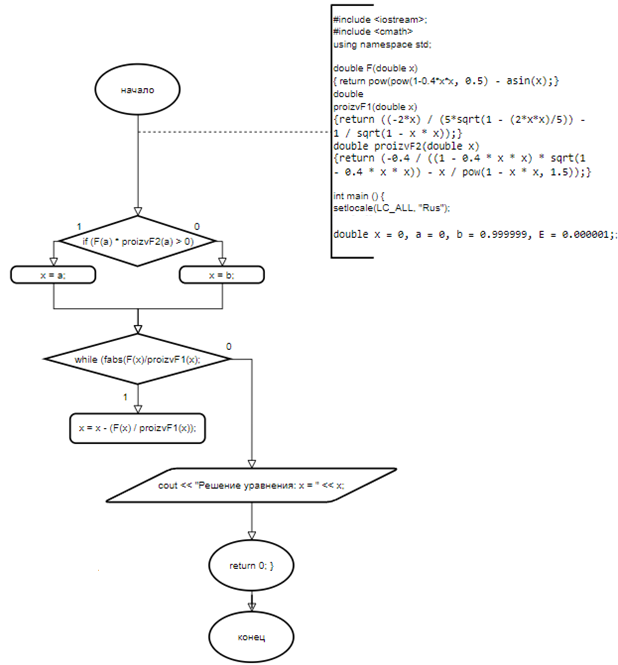
2023**Разработка программы построения равнобедренного треугольника.**

1. **Постановка задачи** – разработать программы для решения нелинейного уравнения тремя методами: половинного деления, Ньютона и итераций.
2. **Блок-схема**

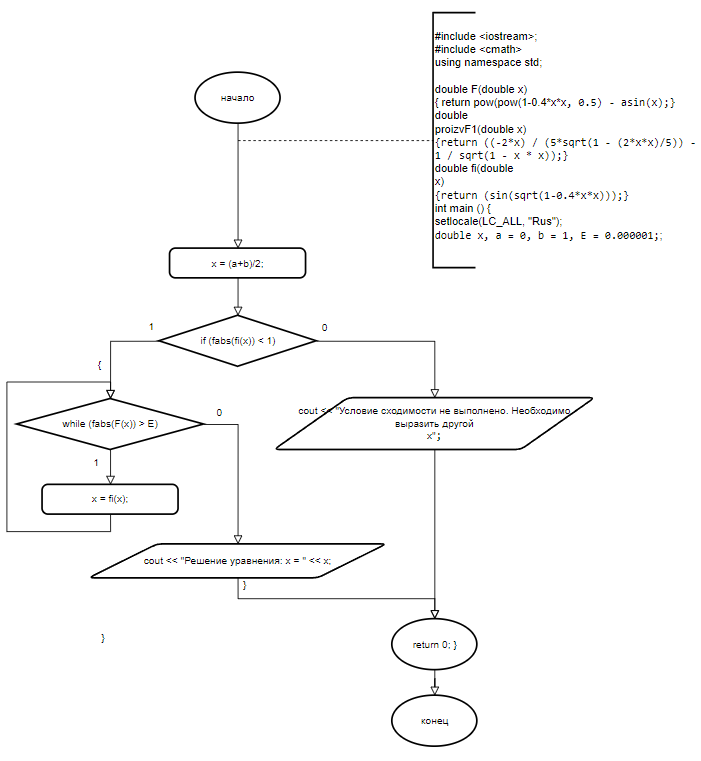
**Метод половинного деления:**

****

**Метод Ньютона:**

****

**Метод итераций:**

****

1. **Листинг программы**

**Метод половинного деления:**

#include <iostream>

using namespace std;

#include <cmath>

double F(double x)

{

return pow(1-0.4\*x\*x, 0.5) - asin(x);

}

int main()

{

setlocale(LC\_ALL, "Rus");

bool flag = 0;

double x0, a = 0, b = 1, E = 0.000001;

while (flag == 0)

{

x0 = (a + b) / 2;

if (F(a) \* F(x0) < 0)

{

b = x0;

}

else a = x0;

if (abs(F(a) \* F(x0)) < E)

{

flag = 1;

}

}

cout << "Решение уравнения: x = " << x0<< "\n";

}

**Метод Ньютона:**

#include <iostream>

using namespace std;

#include <cmath>

double F(double x)

{

return (pow(1-0.4\*x\*x, 0.5) - asin(x));

}

double proizvF1(double x)

{

return ((-2\*x) / (5\*sqrt(1 - (2\*x\*x)/5)) - 1 / sqrt(1 - x \* x));

}

double proizvF2(double x)

{

return (-0.4 / ((1 - 0.4 \* x \* x) \* sqrt(1 - 0.4 \* x \* x)) - x / pow(1 - x \* x, 1.5));

}

int main()

{

setlocale(LC\_ALL, "Rus");

double x = 0, a = 0, b = 0.999999, E = 0.000001;

if (F(a) \* proizvF2(a) > 0)

{

x = a;

}

else x = b;

while (fabs(F(x))>=E)

{

x = x - (F(x) / proizvF1(x));

}

cout << "Решение уравнения: x = " << x<< "\n";

}

**Метод итераций:**

#include <iostream>

using namespace std;

#include <cmath>

double F(double x)

{

return (pow(1-0.4\*x\*x, 0.5) - asin(x));

}

double proizvF1(double x)

{

return ((-2\*x) / (5\*sqrt(1 - (2\*x\*x)/5)) - 1 / sqrt(1 - x \* x));

}

double fi(double x)

{

return (sin(sqrt(1-0.4\*x\*x)));

}

int main()

{

setlocale(LC\_ALL, "Rus");

double x, a = 0, b = 1, E = 0.000001;

x = (a + b) / 2;

if (fabs(fi(x)) < 1)

{

while (fabs(F(x)) > E)

{

x = fi(x);

}

cout << "Решение уравнения: x = " << x<< "\n";

}

else cout << "Условие сходимости не выполнено. Необходимо выразить другой х";

}

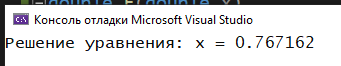
**Результаты выполнения программы**

Точное значение: 0.7672

Вывод программы метода половинного деления:



Вывод программы метода Ньютона:



Вывод программы метода итераций:



Таким образом, с одинаковой заданной точностью самым близким к истинному ответу оказался метод итераций.

Разместим готовые исполнимые файлы в созданный репозиторий на GitHub:

