**Лабораторная работа № 5**

**Компьютерное моделирование на основе нелинейных уравнений**

**Задание**

Написать программу в виде меню, которая реализует решения нелинейных уравнений методами:

1. итерации

2. половинного деления

3. касательных

4. хорд

II. Найти задачу, которая решается через нахождение корней нелинейного уравнения и реализовать компьютерную модель этой задачи

**Задача**

Задача решена посредством программы Visual Studio 2017

**Код программы:**

#include "stdafx.h"

#include <conio.h>

#define \_USE\_MATH\_DEFINES

#include <iostream>

#include <cmath>

using namespace std;

double func(double x)

{

return (sin(M\_PI\*x / 180) - 1 / x);

}

double iteraz(double x, double eps)

{

double rez, f; int i = 0;

cout << "x0= " << x << " ";

do {

rez = f;

f = func(x);

i++;

} while (fabs(rez - f) > eps && i<20000);

cout << i << " итераций" << endl;

return x;

}

double pol\_del(double x0, double x1, double eps)

{

double left = x0, right = x1, x, fl, fr, f;

int i = 0;

cout << "x0= " << x0 << " x1= " << x1 << " ";

do {

x = (left + right) / 2;

f = func(x);

if (f > 0) right = x;

else left = x;

i++;

} while (fabs(f) > eps && i<20000);

cout << i << " итераций" << endl;

return x;

}

double kasat(double x, double eps)

{

double f, df; int i = 0;

cout << "x0= " << x << " ";

do {

f = sin(M\_PI\*x / 180) - 1 / x;

df = M\_PI / 180 \* cos(M\_PI\*x / 180) + 1 / (x\*x);

x = x - f / df;

i++;

} while (fabs(f) > eps && i<20000);

cout << i << " итераций" << endl;

return x;

}

double hord(double x0, double x1, double eps)

{

double rez = x1, f0, f;

int i = 0;

cout << "x0= " << x0 << " x1= " << x1 << " ";

do {

f = sin(M\_PI\*rez / 180) - 1 / rez;

f0 = sin(M\_PI\*x0 / 180) - 1 / x0;

rez = rez - f / (f - f0)\*(rez - x0);

i++;

} while (fabs(f) > eps && i<20000);

cout << i << " итераций" << endl;

return rez;

}

int main()

{

setlocale(LC\_ALL, "Russian");

int count;

cout << "Выберите метод: 1-итераций; 2-половинного деления; 3-касательных; 4-хорд: ";

cin >> count;

switch (count)

{

case 1:

{

cout << iteraz(7, 0.00001);

getch();

return 0;

break;

}

case 2:

{

cout << pol\_del(1.0, 10.0, 0.000001);

getch();

return 0;

break;

}

case 3:

{

cout << kasat(1, 0.00001);

getch();

return 0;

break;

}

case 4:

{

cout << hord(1.0, 10.0, 0.000001);

getch();

return 0;

break;

}

}

}

**Скриншоты**





