**Лабораторная работа №4**

**«Программирование алгоритмов итеративных циклических структур»**

**Задача:** Написать программу вычисляющую корень уравнения x – cos = 0, c точность 0,0001, воспользовавшись формулой (см. исходные данные)

Примечание:

1. Python: Использовать функции и модули.
2. С++: Использовать функции и DLL библиотеки (написать программу в Console, а также в Windows Form).
3. Проверить правильность решения подстановкой найденного корня в уравнение

Результат представить:

1. В виде вывода корня уравнения, вычисленного с данной точностью
2. В виде вывода корня уравнения, вычисленного напрямую

**Исходные данные:**

|  |  |  |
| --- | --- | --- |
| **Формула** | **Значения корня**  **i** | **Начальное значение корня**  **X0** |
| Xi+1=cos(x)i | i=0,1,2,….., | 0 |

**Алгоритм решения задачи:**



**Рис.1.** Общая блок-схема решения задачи



**Рис.2.** Алгоритм функции fun с использованием модуля (Modul)



**Рис.3.** Алгоритм модуля (Modul)

**Код программы:**

from math import\*

import Modul

def fun (x,a,esp):

x=Modul.fun(x,a,esp)

print("Корень уравнения, вычисленный с заданной точностью =",'{0:.14f}'.format(x))

print("Корень уравнения, вычисленный напрямую = ",x1)

x=0

x1=cos(x)

a=1

esp = float(input("Введите точность вычисления = "))

fun(x,a,esp)

exit(0)

**Код модуля (Modul):**

#Модуль

from math import\*

def fun(x,a,esp):

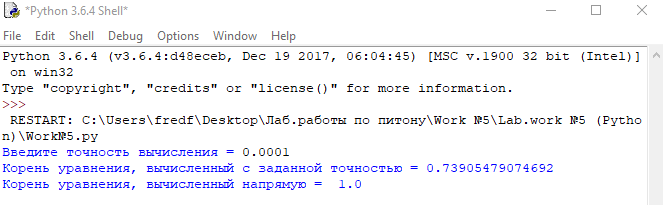
while abs(x-a)>esp:

a=x

x=cos(a)

return x

**Результат работы программы:**



**Рис.4.** Результат работы приложения

**C++**

**Код программы (Console):**

#include "stdafx.h"

#include <iostream>

#include <cmath>

#include "MathFuncsDll.h"

using namespace std;

void fun(double, double, double,double);

int main()

{

setlocale(LC\_ALL, "");

double eps;

double x = 0;

double x1 = cos(x);

double a = 1;

cout << "Введите точность вычисления" << endl;

cin >> eps;

fun(x, a, eps,x1);

system("pause");

return 0;

}

void fun(double x,double a,double eps,double x1)

{

MathFuncs::MyMathFuncs::fun(a, eps, x);

cout << "Корень уравнения с заданной точностью = " << x << endl;

cout << "Корень уравнения, вычисленный напрямую = " << x1 << endl;

}

**Код заголовочного файла (MathFuncsDll.h):**

#pragma once

#ifdef MATHFUNCSDLL\_EXPORTS

#define MATHFUNCSDLL\_API \_\_declspec(dllexport)

#else

#define MATHFUNCSDLL\_API \_\_declspec(dllimport)

#endif

namespace MathFuncs

{

class MyMathFuncs

{

public:

static MATHFUNCSDLL\_API void fun(double a, double eps, double &x);

};

}

**Код библиотеки (MathFuncsDll.cpp):**

#include "stdafx.h"

#include "MathFuncsDll.h"

#include <stdexcept>

#include <cmath>

using namespace std;

namespace MathFuncs

{

void MyMathFuncs::fun(double a, double eps, double & x)

{

while (abs(x - a) > eps)

{

a = x;

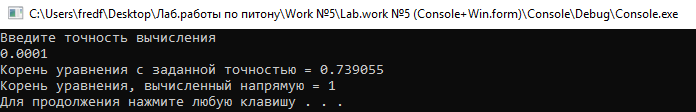
x = cos(a);

}

}

}

**Результат работы программы:**



**Рис.1.** Результат работы консольного приложения

**Код программы (Windows Form):**

#include "MathFuncsDll.h"

#include <cmath>

.

.

.

#pragma endregion

private: System::Void button1\_Click(System::Object^ sender, System::EventArgs^ e) {

double eps;

double x = 0;

double x1 = cos(x);

double a = 1;

eps = Convert::ToDouble(textBox3->Text);

fun(x, a, eps,x1);

}

void fun(double x, double a, double eps,double x1)

{

MathFuncs::MyMathFuncs::fun(a, eps,x);

textBox1->Text = System::Convert::ToString(x);

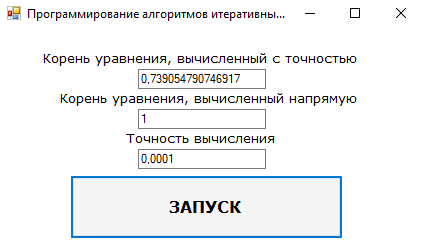
textBox2->Text = System::Convert::ToString(x1);

}

};

}

**Результат работы программы:**



**Рис.1.** Результат работы Windows Form