ЗВІТ

про виконання лабораторної роботи No < 5.3 >

«Функції, що містять розгалуження та цикли з рекурентними співвідношеннями»

з дисципліни

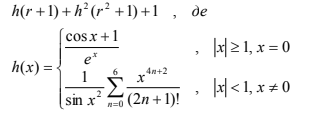
«Алгоритмізація та програмування»

Студента групи КН-106Б

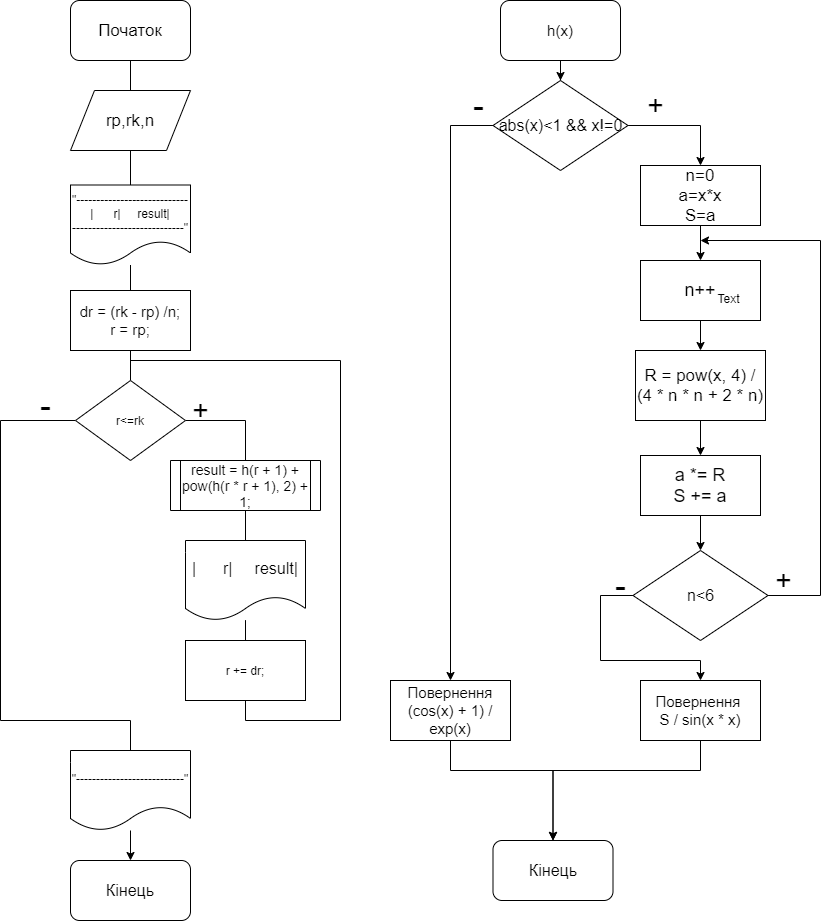
Онишківа Остапа Володимировича

**Мета:** Навчитися використовувати функції.

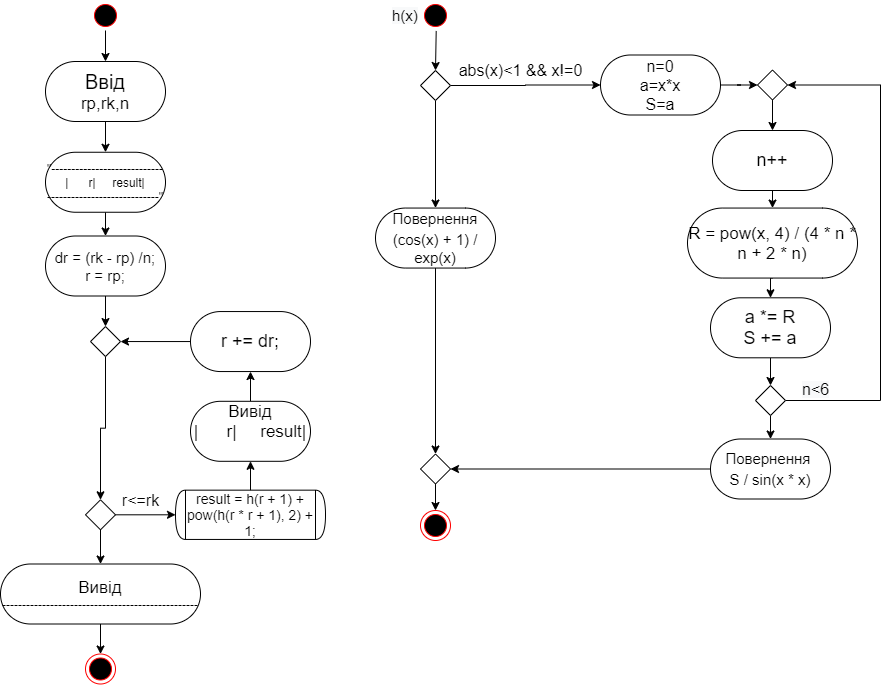
**Умова завдання:** Обчислити і вивести на екран у вигляді таблиці значення виразу від дійсного r на інтервалі від rпоч до rкін з n відрізками розбиття інтервалу. Параметри rпоч, rкін, n вводяться з клавіатури. Таблиця має містити заголовок та шапку. Кожний рядок таблиці має містити значення r та значення виразу. Використовувати допоміжний алгоритм, реалізований за допомогою окремої функції. При обчисленні значення доданків в сумі використовувати рекурентні співвідношення.



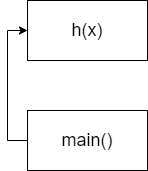
**Блок-схема алгоритму:**



**UML-діаграма алгоритму:**



**Структурна схема:**



**Текст програми:**

// Lab\_5.3.cpp

// < Онишківа Остапа >

// Лабораторна робота No 5.3

// Функції, що містять розгалуження та цикли з рекурентними співвідношеннями.

// Варіант 5

#include<iostream>

#include<iomanip>

#include<cmath>

using namespace std;

double h(const double x);

int main()

{

int n;

double rp, rk,dr,r;

long double result;

cout << "rp = "; cin >> rp;

cout << "rk = "; cin >> rk;

cout << "n = "; cin >> n;

cout << "-------------------------------------------------" << endl;

cout << "|" << setw(7) << "r" << "|"

<< setw(11) << "result" << "|"

<< endl;

cout << "-------------------------------------------------" << endl;

dr = (rk - rp) /n;

r = rp;

while (r<=rk)

{

result = h(r + 1) + pow(h(r \* r + 1), 2) + 1;

cout << "|" << setw(7) << r << "|" << setw(11) <<result << "|" << endl;

r += dr;

}

cout << "-------------------------------------------------" << endl;

return 0;

}

double h(const double x)

{

if (abs(x)<1 && x!=0)

{

double S = 0;

int n = 0;

double a = x \* x;

S = a;

do

{

n++;

double R = pow(x, 4) / (4 \* n \* n + 2 \* n);

a \*= R;

S += a;

} while (n < 6);

return S / sin(x \* x);

}

else

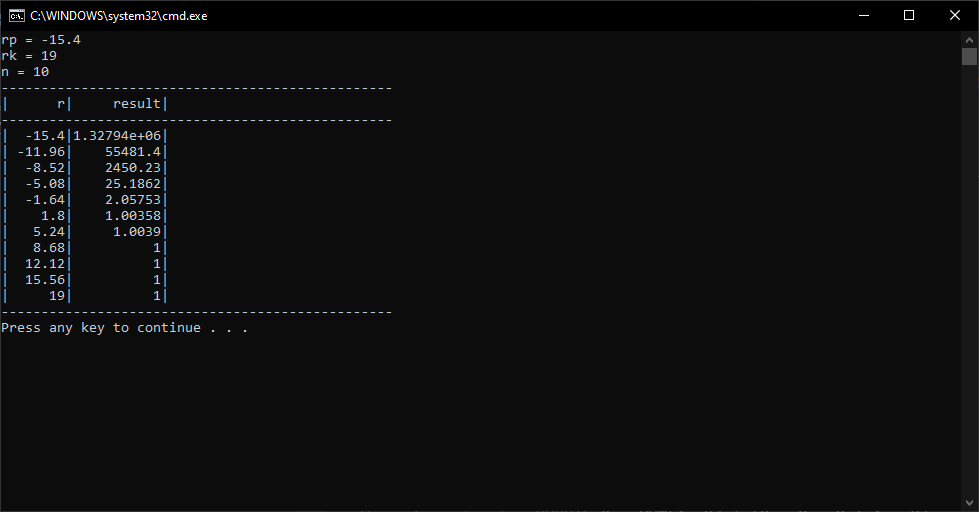
{

return (cos(x) + 1) / exp(x);

}

}

**Результат виконання:**



**Посилання на git-репозиторій з проектом:**

**Текст програми unit-тесту:**

#include "pch.h"

#include "CppUnitTest.h"

#include "../Lab\_5.3/Lab\_5.3.cpp"

using namespace Microsoft::VisualStudio::CppUnitTestFramework;

namespace UnitTest1

{

TEST\_CLASS(UnitTest1)

{

public:

TEST\_METHOD(TestMethod1)

{

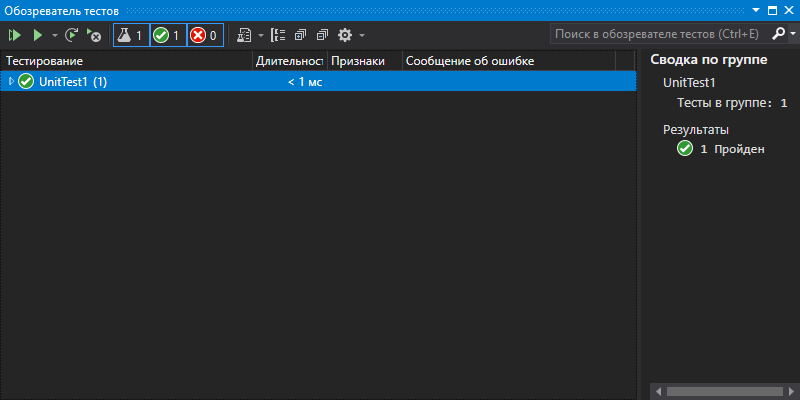
Assert::AreEqual(h(0), 2.);

}

};

}

**Результат unit-тесту:**



**Висновки:** на цій лабораторній роботі я навчився користуватися функціями, малювати структурні схеми програми та робити unit-тести.