ЗВІТ

про виконання лабораторної роботи № 5.4

«Обчислення сум та добутків за допомогою рекурсії»

з дисципліни

«Алгоритмізація та програмування»

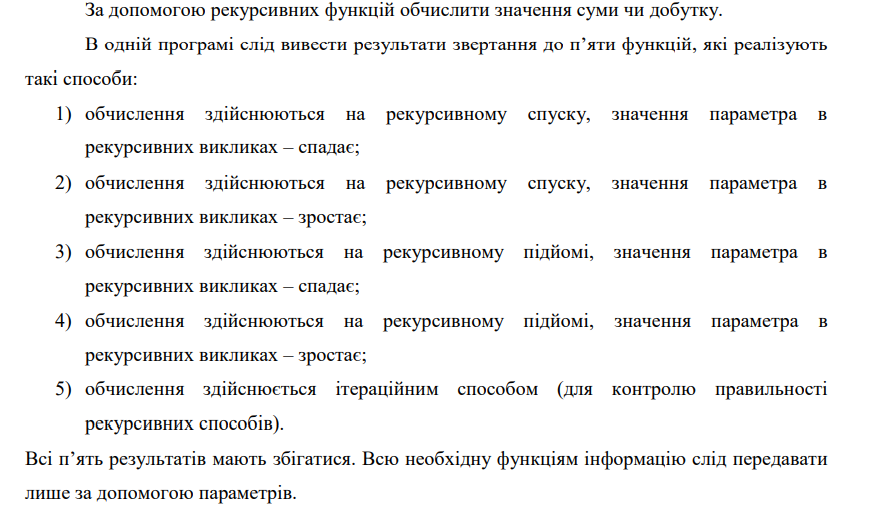
студентки групи РІ-11

Пятницької Вікторії Володимирівни

**Мета роботи**

Навчитися використовувати рекурсивні функції.

**Умова завдання:**

****

****

**Блок-схеми алгоритму програми та кожної функції:**

**UML-діаграма дії програми та кожної функції:**

**Структурна схема програми:**

**Текст програми:**

#include <iostream>

#include <cmath>

using namespace std;

double S0(const int K, const int N)

{

double s = 0;

for (int i = K; i <= N; i++)

s += ((sin(1. \* i)) \* (cos(1. \* i))) / (1 + pow(sin(1.0 \* i), 2));

return s;

}

double S1(const int K, const int N, const int i)

{

if (i > N)

return 0;

else

return ((sin(1. \* i)) \* (cos(1. \* i))) / (1 + pow(sin(1.0 \* i), 2)) + S1(K, N, i + 1);

}

double S2(const int K, const int N, const int i)

{

if (i < K)

return 0;

else

return ((sin(1. \* i)) \* (cos(1. \* i))) / (1 + pow(sin(1.0 \* i), 2)) + S2(K, N, i - 1);

}

double S3(const int K, const int N, const int i, double t)

{

t = t + ((sin(1. \* i)) \* (cos(1. \* i))) / (1 + pow(sin(1.0 \* i), 2));

if (i >= N)

return t;

else

return S3(K, N, i + 1, t);

}

double S4(const int K, const int N, const int i, double t)

{

t = t + ((sin(1. \* i)) \* (cos(1. \* i))) / (1 + pow(sin(1.0 \* i), 2));

if (i <= K)

return t;

else

return S4(K, N, i - 1, t);

}

int main()

{

int K, N;

cout << "K = "; cin >> K;

cout << "N = "; cin >> N;

cout << "(iter) S0 = " << S0(K, N) << endl;

cout << "(rec up ++) S1 = " << S1(K, N, K) << endl;

cout << "(rec up --) S2 = " << S2(K, N, N) << endl;

cout << "(rec down ++) S3 = " << S3(K, N, K, 0) << endl;

cout << "(rec down --) S4 = " << S4(K, N, N, 0) << endl;

return 0;

}

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

**Результати unit-тесту:**

#include "pch.h"

#include "CppUnitTest.h"

#include "../lab\_5.4/lab\_5.4.cpp"

using namespace Microsoft::VisualStudio::CppUnitTestFramework;

namespace UnitTest4

{

TEST\_CLASS(UnitTest4)

{

public:

TEST\_METHOD(TestMethod1)

{

double t;

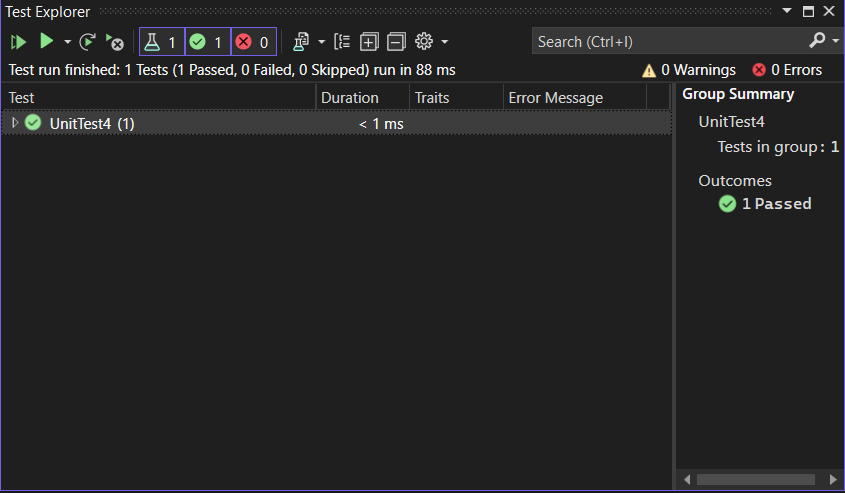
t = S1(4, 1, 4);

Assert::AreEqual (t, 0.0);

}

};

}

****

**Висновки:**Отже, виконавши цю лабораторну роботу я навчилась використовувати рекурсивні функції.