

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЦЕНТРАЛЬНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
МЕХАНІКО-ТЕХНОЛОГІЧНИЙ ФАКУЛЬТЕТ  
КАФЕДРА КІБЕРБЕЗПЕКИ ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

ЗВІТ  
з лабораторної роботи № 8  
з дисципліни  
«Базові методології та технології програмування»  
на тему:  
РЕАЛІЗАЦІЯ СТАТИЧНИХ БІБЛІОТЕК МОДУЛІВ ЛІНІЙНИХ  
ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ

Виконав:  
студент академічної групи КН-22  
ПІВНЕНКО О.М.

Перевірив:  
викладач  
СОБІНОВ О.Г.

## Лабораторна робота № 8

### Варіант № 19

#### ТЕМА: РЕАЛІЗАЦІЯ СТАТИЧНИХ БІБЛІОТЕК МОДУЛІВ ЛІНІЙНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ

**МЕТА:** набуття ґрунтовних вмінь і практичних навичок реалізації метода функціональної декомпозиції задач, метода модульного (блочного) тестування, представлення мовою програмування C/C++ даних скалярних типів, арифметичних і логічних операцій, потокового введення й виведення інформації, розроблення програмних модулів у кросплатформовому середовищі Code::Blocks (GNU GCC Compiler).

#### ЗАВДАННЯ:

8.1 За значеннями  $x$ ,  $y$ ,  $z$  обчислюється  $S$ .

$$S = (2z + 1)^x - \sqrt{\left| y - \frac{1}{2}z \right|} + z + \pi$$

8.2 За послідовними запитами вводяться числа  $x$ ,  $y$ ,  $z$  та символи  $a$ ,  $b$ . Вивести (включити у потік STL – cout):

8.2.1. Прізвище та ім'я розробника програми зі знаком охорони авторського права «(C)» (від англ. copyright)

8.2.2. Результат логічного виразу в текстовому вигляді ( false/true ):

$$a + 1 \geq b ?$$

8.2.3 Значення  $x$ ,  $y$ ,  $z$  в десятковій і шістнадцятковій системах числення;  $S$ , що обчислюється функцією `s_calculation()` заголовкового файлу `ModulesПрізвище.h`

## ХІД ЛАБОРАТОРНОЇ РОБОТИ

### Завдання 1

Текст програми зазначений у додатку 1. Блок-схема до коду зазначена на рисунку 8.1. Вихідний код програми TestDriver зазначений у додатку 2. Тест-сьют зазначений у додатку 3.

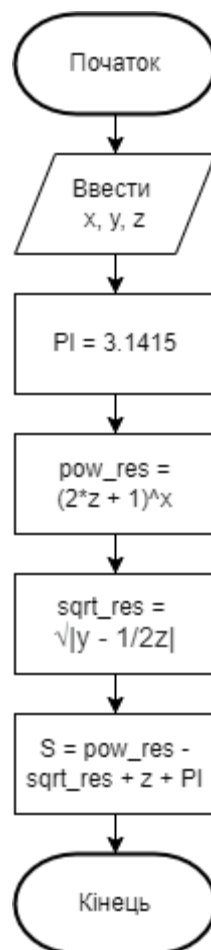


Рисунок 8.1. Блок-схема до лістингу 8.1.

### Завдання 2

Вихідний код додатку наявний у лістингу 8.2, блок-схеми алгоритмів наведені на рисунках 8.2, 8.3, 8.4. Тест-сьют наведений у додатку 4.

### Лістинг 8.2

```

#include <iostream>
#include <windows.h>

using namespace std;

void HexDecimal(int x, int y, int z)
{

```

```

    cout << "X in Decimal: " << dec << x << endl;
    cout << "X in Hex: " << hex << uppercase << x << endl;

    cout << "Y in Decimal: " << dec << y << endl;
    cout << "Y in Hex: " << hex << uppercase << y << endl;

    cout << "Z in Decimal: " << dec << z << endl;
    cout << "Z in Hex: " << hex << uppercase << z << endl;
}

bool BoolExpression(int a, int b)
{
    if(a + 1 >= b)
        return true;
    else
        return false;
}

void SoftwareDevelop(void)
{
    cout << "-----\n";
    cout << "-Розробник: Півненко Олександр-----\n";
    cout << "-Це застосунок для обробки масиву--\n";
    cout << "-----© All Rights Reserved-----\n";
}

int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);

    int a = 0; int b = 0; int x, y, z = 0;

    SoftwareDevelop();
    cout << endl;

    cout << "Уведіть, будь ласка, числа x, y, z." << endl;
    cin >> x; cin >> y; cin >> z;

    HexDecimal(x, y, z);
    cout << endl;

    cout << "Уведіть, будь ласка, числа a i b." << endl;
    cin >> a;
    cin >> b;
}

```

```

    if(BoolExpression(a, b) == true)
        cout << "Bool Expression " << dec << "( " << a << ", " << b << " )"
    << " is true";
    else
        cout << "Bool Expression " << dec << "( " << a << ", " << b << " )"
    << " is false";

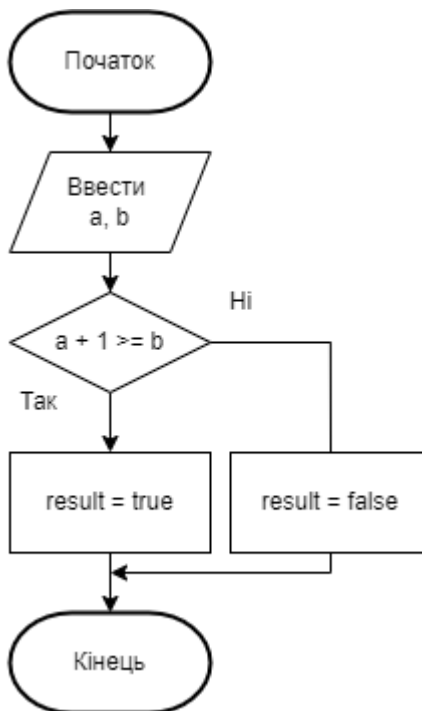
    getchar();
    getchar();

    return 0;
}

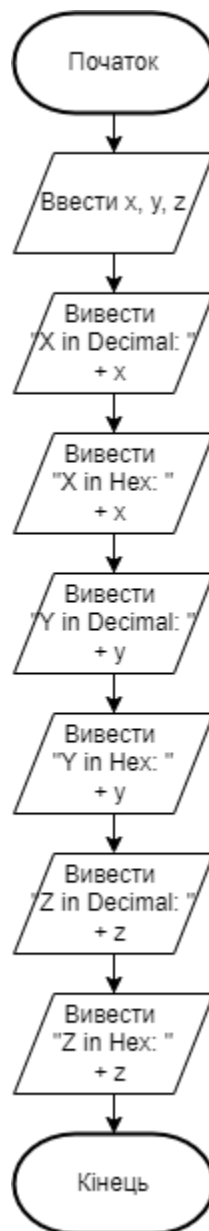
```



**Рисунок 8.2. Блок-схема алгоритму до завдання 8.2.1.**



**Рисунок 8.3. Блок-схема алгоритму до завдання 8.2.2.**



**Рисунок 8.4. Блок-схема алгоритму до завдання 8.2.3**

## **ВИСНОВОК**

У ході виконання лабораторної роботи я закріпив свої знання ( а також здобув нові ) з мови програмування C++. Попрацював, а також вдосконалив свої навички роботи із текстовими редакторами Sublime Text, Notepad++, IDE Code::Blocks, IDE Visual Studio. Мав змогу також створити свою першу статичну бібліотеку. Закріпив знання із тестування. Створив два тест-сьюта. Провів тестування вихідних додатків. Попрацював із заголовковими файлами у C++. Створив програму для обчислення математичного виразу із використанням процедур для обчислювання модуля, квадратного кореня, піднесення до степеня. Використав псевдографіку у своїх додатках (виведення інформації про

розробника). Згадав матеріал із комп'ютерної логіки ( шістнадцяткова, вісімкова, десяткові системи числення ). Не можна не сказати й про власні враження від програмування на C++: я отримав превелике задоволення. Нарешті, увесь функціонал програми не потрібно реалізовувати самотужки: у плюсах є дуже багато вбудованих засобів, які суттєво облегшують процес роботи. А те, що було у С, стало зручнішим ( наприклад, консольне введення/виведення за допомогою cin/cout).

```
#include <math.h>

#define PI 3.1415

float s_calculation(int x, int y, int z){

    float pow_res = pow(2*z + 1, x);

    float sqrt_res = sqrt(abs(y - ((float)1/(float)2) * z));

    float S = pow_res - sqrt_res + z + PI;

    return S;

}
```



```
#include <iostream>

#include <iomanip>

#include <windows.h>

#include <math.h>

#include "ModulesPivnenko.hpp"

using namespace std;

void CheckResults(bool input){

    if(input == true)

        cout << "PASSED";

    else

        cout << "FAILED";

}

bool CompareResults(float a, float b){

    int i_a = a * pow(10, 2);

    int i_b = b * pow(10, 2);

    if(i_a == i_b)

        return true;

    else

        return false;

}

void TestCase(int x, int y, int z, int CaseNum, float Etalon){

    cout << endl << CaseNum << "-й тест-кейс." << endl;

    float FirstTest = s_calculation(x, y, z);

    cout << "Значення еталонне та отримане = " << fixed << setprecision(4)
    << FirstTest << ":" << Etalon << endl;
```

```
bool CompRes = CompareResults(FirstTest, Etalon);

CheckResults(CompRes);

}

int main()

{

    SetConsoleCP(1251);

    SetConsoleOutputCP(1251);


    TestCase(1, 1, 1, 1, 6.4344);

    TestCase(2, 1, 1, 2, 12.4344);

    TestCase(2, 3, 2, 3, 28.7273);

    TestCase(2, -2, 2, 4, 28.4095);

    TestCase(0, 0, 0, 5, 4.1415);

    TestCase(-1, 0, 0, 6, 4.1415);

    TestCase(5, 1, 1, 7, 246.4344);

    TestCase(10, 1, 1, 8, 59052.4336);

    TestCase(3, 3, 3, 9, 347.9168);

    TestCase(7, 5, 1, 10, 2189.0203);


    getchar();

    return 0;

}
```

## Додаток 3.

Artifact: Test Suite

Date: 3/3/2023

Назва тестового набору Test Suite Description	TS_lab8_1
Назва проекту / ПЗ Name of Project / Software	ModulesPivnenko
Рівень тестування Level of Testing	системний / System Testing
Автор тест-сьюта Test Suite Author	Олександр Півненко
Виконавець Implementer	Олександр Півненко

Ід-р тест- кейса / Test Case ID	Дії (кроки) / Action (Test Steps)	Очікуваний результат / Expected Result	Результат тестування / Test Result
TC_01	1.Відкрити файл TestDriver.exe	Перед користувачем з'явилось вікно консолі. Текст, який зображений: «1-й тест-кейс. Значення еталонне та отримане = 6.4344:6.4344. PASSED».	PASSED
TC_02	1.Відкрити файл TestDriver.exe	Перед користувачем з'явилось вікно консолі. Текст, який зображений: «2-й тест-кейс. Значення еталонне та отримане = 12.4344:12.4344 PASSED».	PASSED
TC_03	1.Відкрити файл TestDriver.exe	Перед користувачем з'явилось вікно консолі. Текст, який зображений: «3-й тест-кейс. Значення еталонне та отримане = 28.7273:28.7273 PASSED».	PASSED
TC_04	1.Відкрити файл TestDriver.exe	Перед користувачем з'явилось вікно консолі. Текст, який зображений: «4-й тест-кейс. Значення еталонне та отримане = 28.4095:28.4095 PASSED».	PASSED

TC_05	1.Відкрити файл TestDriver.exe	Перед користувачем з'явилося вікно консолі. Текст, який зображений: «Значення еталонне та отримане = 4.1415:4.1415 PASSED».	PASSED
TC_06	1.Відкрити файл TestDriver.exe	Перед користувачем з'явилося вікно консолі. Текст, який зображений: «2-й тест-кейс. Значення еталонне та отримане = 4.1415:4.1415 PASSED».	PASSED
TC_07	1.Відкрити файл TestDriver.exe	Перед користувачем з'явилося вікно консолі. Текст, який зображений: «7-й тест-кейс. Значення еталонне та отримане = 246.4344:246.4344 PASSED».	PASSED
TC_08	1.Відкрити файл TestDriver.exe	Перед користувачем з'явилося вікно консолі. Текст, який зображений: «8-й тест-кейс. Значення еталонне та отримане = 59052.4336:59052.4336 PASSED».	PASSED
TC_09	1.Відкрити файл TestDriver.exe	Перед користувачем з'явилося вікно консолі. Текст, який зображений: «9-й тест-кейс. Значення еталонне та отримане = 347.9168:347.9168 PASSED».	PASSED
TC_10	1.Відкрити файл TestDriver.exe	Перед користувачем з'явилося вікно консолі. Текст, який зображений: «10-й тест-кейс. Значення еталонне та отримане = 2189.0203:2189.0203 PASSED».	PASSED

## Додаток 4.

Artifact: Test Suite

Date: 3/3/2023

Назва тестового набору Test Suite Description	TS_lab8_2
Назва проекту / ПЗ Name of Project / Software	Pivnenko-task.exe
Рівень тестування Level of Testing	системний / System Testing
Автор тест-сьюта Test Suite Author	Олександр Півненко
Виконавець Implementer	Олександр Півненко

Ід-р тест- кейса / Test Case ID	Дії (кроки) / Action (Test Steps)	Очікуваний результат / Expected Result	Результат тестування / Test Result
TC_01	1.Відкрити файл Pivnenko-task.exe.	Перед користувачем з'явилося вікно консолі. Текст, який зображений: «----- -Розробник: Півненко Олександр----- -Це застосунок для обробки масиву-- -----© All Rights Reserved----- .»	PASSED
TC_02	1.Відкрити файл Pivnenko-task.exe.	Перед користувачем з'явилося вікно консолі. Текст, який зображений: «Уведіть, будь ласка, числа x, y, z.».	PASSED
TC_03	1.Відкрити файл Pivnenko-task.exe. 2.Увести 27, 15, 17	Перед користувачем з'явилося вікно консолі. Текст, який зображений: «Уведіть, будь ласка, числа x, y, z. X in Decimal: 27 X in Hex: 1B Y in Decimal: 15 Y in Hex: F Z in Decimal: 17 Z in Hex: 11».	PASSED
TC_04	1.Відкрити файл Pivnenko-task.exe. 2.Увести 5 4 3.	Перед користувачем з'явилося вікно консолі. У ньому наявне «Уведіть, будь ласка, числа x, y, z. 5 4 3	PASSED

		X in Decimal: 5 X in Hex: 5 Y in Decimal: 4 Y in Hex: 4 Z in Decimal: 3 Z in Hex: 3»	
TC_05	1.Відкрити файл Pivnenko-task.exe. 2.Увести 8 8 8 8 8	Перед користувачем з'явилося вікно консолі. У ньому наявне «Уведіть, будь ласка, числа x, y, z. 8 8 8 8 8 X in Decimal: 8 X in Hex: 8 Y in Decimal: 8 Y in Hex: 8 Z in Decimal: 8 Z in Hex: 8»	PASSED
TC_06	1.Відкрити файл Pivnenko-task.exe. 2.Увести 8 8 8 3. Увести 9 10	Перед користувачем з'явилося вікно консолі. У ньому наявне «Уведіть, будь ласка, числа a і b. 9 10 Bool Expression ( 9, 10 ) is true»	PASSED
TC_07	1.Відкрити файл Pivnenko-task.exe. 2.Увести 8 8 8 3. Увести 5 10	Перед користувачем з'явилося вікно консолі. У ньому наявне «Уведіть, будь ласка, числа a і b. 5 10 Bool Expression ( 5, 10 ) is false»	PASSED
TC_08	1.Відкрити файл Pivnenko-task.exe. 2.Увести 8 8 8 3. Увести -245 100	Перед користувачем з'явилося вікно консолі. У ньому наявне «Уведіть, будь ласка, числа a і b. -245 100 Bool Expression ( -245, 100 ) is false»	PASSED