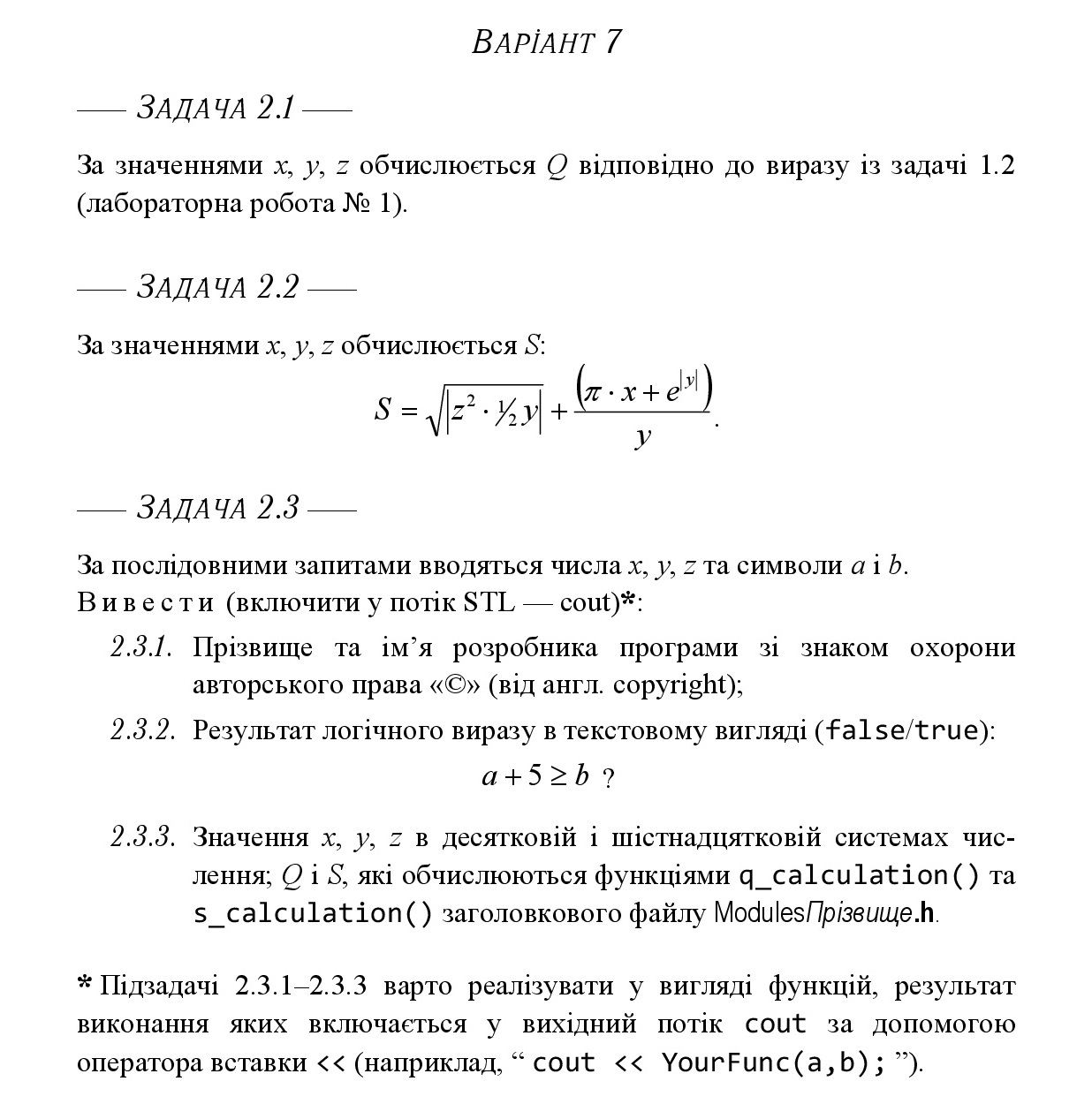
**Мета лабораторної роботи**

Полягає у набутті ґрунтовних вмінь і практичних навичок застосування теоретичних положень методології модульного програмування, реалізації метода функціональної декомпозиції задач, метода модульного (блочного) тестування, представлення мовою програмування С/С++ даних скалярних типів, арифметичних і логічних операцій, потокового введення й виведення інформації, розроблення програмних модулів та засобів у кросплатформовому середовищі Code::Blocks (GNU GCC Compiler).

**Завдання до лабораторної роботи**

1. Реалізувати статичну бібліотеку модулів libModulesПрізвище C/C++, яка містить функції розв’язування задач 2.1 та 2.2.

2. Реалізувати програмне забезпечення розв’язування задачі 2.3 — консольний додаток.

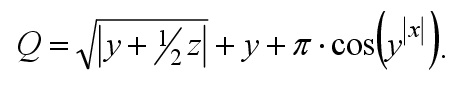
****

**Задача 2.1**

Аналіз задачі:

Дана задача має виконувати наступні функції:

-При введенні значення x, y, z виконується обчислення значення Q за формулою:

****

Строга постановка задачі:

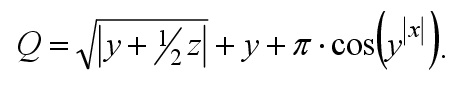
Вихідні дані: змінна result;

Вхідні дані: змінні x,y,z;

Аналіз вимог до програмного забезпечення:

- Модуль перевіряє вхідні дані на валідність.

- Функція повинна обчислювати формулу:

****

Проектування архітектури програмного забезпечення:

Даний модуль включає в себе перевіряючу частину, а також робочу частину.

Детальне проектування програмного забезпечення:

Перевіряюча чистина має виконувати перевірку вхідних даних на валідність.

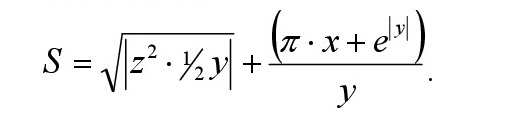
Робоча частина має виконувати необхідні обчислення.

**Задача 2.2**

Аналіз задачі:

Дана задача має виконувати наступні функції:

* При введенні значення x, y, z виконується обчислення значення S за формулою:



Строга постановка задачі:

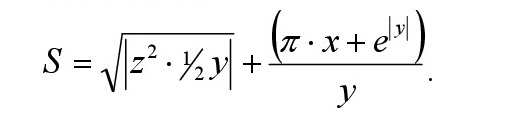
Вихідні дані: змінна result1;

Вхідні дані: змінні x, y, z;

Обмеження та допущення: уся введена і виведена інформація є типом Float; Змінна y не може дорівнювати нулю.

Аналіз вимог до програмного забезпечення:

* Модуль перевіряє вхідні дані на валідність.
* Функція повинна обчислювати формулу:



Проектування архітектури програмного забезпечення:

Даний модуль включає в себе перевіряючу частину, а також робочу частину.

Детальне проектування програмного забезпечення:

Перевіряюча чистина має виконувати перевірку вхідних даних на валідність.

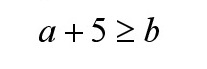
Робоча частина має виконувати необхідні обчислення.

**Задача 2.3**

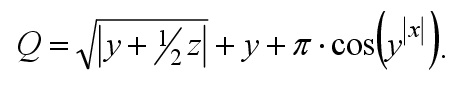
Аналіз задачі:

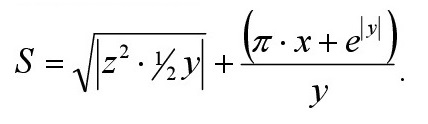
Дана задача має виконувати наступні функції:

* Прізвище, ім’я, по-батькові розробника, а також знак охорони авторського права.
* При введені параметрів a та b обчислюється логічний вираз:

****

* При введенні значення x, y, z виконується обчислення значення S за формулами:





Строга постановка задачі:

Вихідні дані: потокове виведення значення логічного виразу, а також результатів обчислення задач 2.1 та 2.2;

Вхідні дані: змінні x, y, z, a, b;

Обмеження та допущення: відсутні.

Аналіз вимог до програмного забезпечення:

- Програма має містити інформацію про автора, а також знак захисту інформації.

-Програма повинна виводити результат логічного виразу у числовому форматі, а також результати обчислень функцій 2.1 та 2.2 у десятковій та шістнадцядковій системах числення.

Проектування архітектури програмного забезпечення:

## Дана програма включає в себе робочу частину,підключений модуль з функціями q\_calcunation і s\_calculation, а також перевіряючу частину.

Детальне проектування програмного забезпечення:

Вихідний код умовно розділений на три частини(вивід інформації про розробника, вивід результату обчислення логічного виразу та вивід результатів обчислення задач 2.1 та 2.2).

Вихідний код проекта TestDriver //2

#include <iostream>

#include <ModuleAbashina.h>

#include <stdlib.h>

using namespace std;

const char num[13] = {'-',',','.','0','1','2','3','4','5','6','7','8','9'};

bool isNum (string s);

double toNum(string s);

int main()

{

setlocale(LC\_CTYPE,"ukr");

string ix, iy, iz, ia, ib;

double x ,y, z, a, b;

bool repeat = true;

char exit;

cout << name\_abashina();

while (repeat)

{

cout << "Введiть число х: ";

cin >> ix;

if(isNum(ix)) x = toNum(ix);

else { cout << "Помилка вводу даних. Очiкувалось число. Спробуйте ще раз" << endl; continue;}

cout << "Введiть число у: ";

cin >> iy;

if(isNum(iy)) y = toNum(iy);

else { cout << "Помилка вводу даних. Очiкувалось число. Спробуйте ще раз" << endl; continue;}

cout << "Введiть число z: ";

cin >> iz;

if(isNum(iz)) z = toNum(iz);

else { cout << "Помилка вводу даних. Очiкувалось число. Спробуйте ще раз" << endl; continue;}

cout << "Введiть число а: ";

cin >> ia;

if(isNum(ia)) a = toNum(ia);

else { cout << "Помилка вводу даних. Очiкувалось число. Спробуйте ще раз" << endl; continue;}

cout << "Введiть число b: ";

cin >> ib;

if(isNum(ib)) b = toNum(ib);

else { cout << "Помилка вводу даних. Очiкувалось число. Спробуйте ще раз" << endl; continue;}

cout << endl << "Result \"a + 5 >= b\" is "<< task\_2(a,b);

cout << q\_calculation(x,y,z) << endl;

cout << s\_calculation(x,y,z) << endl;

cout << endl << "Для повтору натиснiть клавiшу Q/q, для виходу з програми натиснiть будь яку (iншу) клавiшу " << endl;

cin >> exit;

if (exit == 'q' || exit == 'Q')

{

repeat = false;

}

}

return 0;

}

double toNum(string s)

{

for (int i = 0; i < s.length(); i++)

{

if (s[i] == ',') s[i] = '.';

}

return atof(s.c\_str());

}

bool isNum (string s)

{

bool debug;

if (s.length() == 1 && (s[0] == ',' || s[0] == '.'))

{

return false;

}

bool b = false;

for (int i = 0; i < s.length(); i++)

{

b = false;

for (int j = 0; j < 13; j++)

{

if(s[i] == num[j])

{

b = true;

break;

}

}

if (!b)

{

return false;

}

}

return true;

}

Додаток №1

Тест-сьют для модульного тестування

|  |  |
| --- | --- |
| Назва тестового набору  Test Suite Description | TS\_lab2\_Modules |
| Назва проекта / ПЗ  Name of Project / Software | libModulesAbashina.a |
| Рівень тестування  Level of Testing | модульне / Unit Testing |

| Ід-р тест-кейса | Вхідні дані | Очікуваний  результат | Вихідні дані | Результат тестування  (passed/failed/ blocked) |
| --- | --- | --- | --- | --- |
| Test\_Q1 | 1, 2, 3 | Q(16) = 2.903F3B7A;\tQ(10) = 2.56346 | Q(16) = 2.903F3B7A;\tQ(10) = 2.56346 | **passed** |
| Test\_Q2 | 1, 1, 1 | Q(16) = 3.EC125359;\tQ(10) = 3.92215 | Q(16) = 3.EC125359;\tQ(10) = 3.92215 | **passed** |
| Test\_Q3 | 0, -1, 0 | Q(16) = 1.B2897218;\tQ(10) = 1.69741 | Q(16) = 1.B2897218;\tQ(10) = 1.69741 | **passed** |
| Test\_Q4 | 10, 10, 10 | Q(16) = 10.9DB02DFF;\tQ(10) = 16.616 | Q(16) = 10.9DB02DFF;\tQ(10) = 16.616 | **passed** |
| Test\_Q5 | 1, 5, 10 | Q(16) = 9.0DAD80CF;\tQ(10) = 9.05343 | Q(16) = 9.0DAD80CF;\tQ(10) = 9.05343 | **passed** |

| Ід-р тест-кейса | Вхідні дані | Очікуваний  результат | Вихідні дані | Результат тестування  (passed/failed/ blocked) |
| --- | --- | --- | --- | --- |
| Test\_S1 | 1, 2, 3 | S(16) = 7.80301339;\tS = 7.50073 | S(16) = 7.80301339;\tS = 7.50073 | **passed** |
| Test\_S2 | 1, 1, 1 | S(16) = 6.9125AF1F;\tS = 6.56698 | S(16) = 6.9125AF1F;\tS = 6.56698 | **passed** |
| Test\_S3 | 0, -1, 0 | S(16) = .00000000;\tS = -1 | S(16) = .00000000;\tS = -1 | **passed** |
| Test\_S4 | 10, 10, 10 | S(16) = 8D0.6C55E76E;\tS = 2256.42 | S(16) = 8D0.6C55E76E;\tS = 2256.42 | **passed** |
| Test\_S5 | 1, 5, 10 | S(16) = 13.7F239F97;\tS = 19.4966 | S(16) = 13.7F239F97;\tS = 19.4966 | **passed** |

Додаток №2

Результат виконання тестового драйвера

Test task 1

TestCase#1 Passed

TestCase#2 Passed

TestCase#3 Passed

TestCase#4 Passed

TestCase#5 Passed

Test task 2

TestCase#1 Passed

TestCase#2 Passed

TestCase#3 Passed

TestCase#4 Passed

TestCase#5 Passed

Додаток №3

Лістинг текстового драйвера

string qtest [5];

qtest[0] = q\_calculation(1, 2, 3);

qtest[1] = q\_calculation(1, 1, 1);

qtest[2] = q\_calculation(0, -1, 0);

qtest[3] = q\_calculation(10, 10, 10);

qtest[4] = q\_calculation(1, 5, 10);

cout << " Test task 1 " << endl;

if (qtest[0] == "Q(16) = 2.903F3B7A;\tQ(10) = 2.56346" )

cout << "TestCase#1 Passed" << endl;

if (qtest[1] == "Q(16) = 3.EC125359;\tQ(10) = 3.92215")

cout << "TestCase#2 Passed" << endl;

if (qtest[2] == "Q(16) = 1.B2897218;\tQ(10) = 1.69741")

cout << "TestCase#3 Passed" << endl;

if (qtest[3] == "Q(16) = 10.9DB02DFF;\tQ(10) = 16.616")

cout << "TestCase#4 Passed" << endl;

if (qtest[4] == "Q(16) = 9.0DAD80CF;\tQ(10) = 9.05343")

cout << "TestCase#5 Passed" << endl;

string s1, s2, s3, s4, s5;

string stest1, stest2, stest3, stest4, stest5;

s1 = s\_calculation(1, 2, 3);

s2 = s\_calculation(1, 1, 1);

s3 = s\_calculation(0, -1, 0);

s4 = s\_calculation(10, 10, 10);

s5 = s\_calculation(1, 5, 10);

cout << " Test task 2 " << endl;

if (s1 == "S(16) = 7.80301339;\tS = 7.50073")

cout << "TestCase#1 Passed" << endl;

if (s2 == "S(16) = 6.9125AF1F;\tS = 6.56698")

cout << "TestCase#2 Passed" << endl;

if (s3 == "S(16) = .00000000;\tS = -1")

cout << "TestCase#3 Passed" << endl;

if (s4 == "S(16) = 8D0.6C55E76E;\tS = 2256.42")

cout << "TestCase#4 Passed" << endl;

if (s5 == "S(16) = 13.7F239F97;\tS = 19.4966")

cout << "TestCase#5 Passed" << endl;

Додаток №4

Лістинг створеного модуля

#include<iostream>

#include<cstring>

#include<cmath>

#include<clocale>

#include <math.h>

#include <sstream>

#include <cstdlib>

using namespace std;

const double pi = 3.1415926535897932384626433832795;

string name\_abashina()

{

return "Абашина Анастасiя Андрiївна © \n";

}

string task\_2 (int a, int b)

{

if (a+5 >= b)

{

return "true \n";

} else

return "false \n";

}

string doubleToHex (double input)

{

string digits[16] = { "0", "1", "2", "3", "4", "5", "6", "7", "8", "9", "A", "B", "C", "D", "E", "F" };

string hex;

int int\_num = input, buffer;

double frac\_num = input - int\_num;

//cout << int\_num << endl << frac\_num << endl;

while (int\_num>0)

{

hex.insert(0, digits[int\_num % 16]);

int\_num /= 16;

}

hex += ".";

for (int i = 0; i < 8; i++)

{

frac\_num \*= 16;

buffer = floor(frac\_num);

hex += digits[buffer];

frac\_num -= floor(frac\_num);

}

return hex;

}

string q\_calculation (double x, double y, double z)

{

try

{

double q;

q=(sqrt(abs(y+z/2))+y+pi\*cos(pow(y, abs(x))));

std::ostringstream strs;

strs << q;

std::string str = strs.str();

if (q != q) // неопределенность (NaN) не равна ни одному числу, И НЕ РАВНА САМОЙ СЕБЕ

{

return "Невiрно введенi даннi";

}

else

{

return "Q(16) = " + doubleToHex(q) + ";\tQ(10) = " + str;

}

}

catch(const exception&error)

{

return "Невiрно введенi даннi";

}

}

string s\_calculation(double x, double y, double z)

{

try

{

double s;

s = sqrt(abs(z\*z\*(y/2)))+((pi\*x+(exp(abs(x)))/y));

std::ostringstream strs;

strs << s;

std::string str = strs.str();

if (isinf(s))

{

return "Невiрно введенi даннi1";

}

return "S(16) = " + doubleToHex(s) + ";\tS = "+ str;

}

catch(const exception&error)

{

return "Невiрно введенi даннi2";

}

}

**Висновок**

Перед тим як виконати лабораторну роботу №2 я ознайомилась с теоретичними відомостями, підготувала відповіді на контрольні питання та опрацювала самостійну підготовку до виконанні лабораторної роботи №2.

До кожної здачі лабораторної роботи, я спочатку виконала аналіз поставленої задачі, і чітко встановила вхідні і вихідні данні а також обмеження і допущення де вони потрібні.

У ході виконання лабораторної роботи я навчилася створювати та використовувати модулі на мові с++. Крім цього я також навчилася створювати статичні бібліотеки та підключати їх до свого проекта.

Завдання цієї лабораторної роботи допомогли мені ознайомитись з модульним підходом реалізацій ПЗ.