Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

ЗВІТ

ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 8

з навчальної дисципліни

“Базові методології та технології програмування”

Реалізація статичних бібліотек модулів лінійних обчислювальних процесів

ЗАВДАННЯ ВИДАВ

доцент кафедри кібербезпеки та програмного забезпечення

Доренський О. П.

[https://github.com/odorenskyi/](https://github.com/odorenskyi/Dmytro-Parkhomenko-KB18)

ВИКОНАВ

студент академічної групи КБ-21

Кобізь Андрій

ПЕРЕВІРИВ

ст. викладач кафедри кібербезпеки   
та програмного забезпечення

Усик П. С.

Кропивницький – 2021

Лабораторна робота №8

Варіант №20

*Мета роботи* полягає у набутті ґрунтовних вмінь і практичних навичок 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](http://www.codeblocks.org/) (GNU GCC Compiler).

Завдання 8.1

В даному завданні потрібно за введеними числами знайти S по формулі: S = *(x+y+z)^2-2\*3.14\*sqrt(z/3)*

Тож для початку треба розробити алгоритм модулю виконання даного завдання.

Початок

1. Ввести x;
2. Ввести y;
3. Ввести z;
4. Визначити S за формулою: *(x+y+z)^2-2\*3.14\*sqrt(z/3)*

Кінець

Таким чином буде виглядати алгоритм для модуля. Було створено статичну бібліотеку та заголовковий файл.

Також розроблено контрольні приклади (див. ДОДАТОК А).

За допомогою контрольних прикладів було створено тестовий драйвер, що довів справність коду:

*Testcase #1 is Passed.*

*Testcase #2 is Passed.*

*Testcase #3 is Passed.*

*Testcase #4 is Passed.*

*Testcase #5 is Passed.*

*Testcase #6 is Passed.*

*Testcase #7 is Passed.*

*Testcase #8 is Passed.*

*Testcase #9 is Passed.*

*Testcase #10 is Passed.*

Всі отримані результати збіглися з очікуваними.

Завдання 8.2

В даному завданні потрібно послідовно вводити 3 числа та 2 символи та за допомогою розроблених функцій виконати деякі дії з введеними числами та символами. Програмний код матиме 3 функції: Відображення інформації про розробника; Результат логічного виразу; Значення чисел обрахованих за допомогою функції *s\_calculation* в різних системах числення.

Потрібно створити алгоритм для результату логічного виразу:

Початок

1.Ввести a та b;

2.Якщо a + 1 < b, то результат дорівнює 1, перейти на крок 4;

3. Результат дорівнює 0;

4. Відобразити результат;

Кінець

Для двох інших функцій алгоритм створювати не обов’язково.

Додатково було створено тестовий набір для проведення системного тестування(див. ДОДАТОК Б)

Висновок:

*Мета роботи* була у набутті ґрунтовних вмінь і практичних навичок 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](http://www.codeblocks.org/) (GNU GCC Compiler). До завдання ретельно підготувався завчасно. Довелось вивчити декілька питаннь, через які могли б виникнути деякі проблеми під час виконання лабораторної роботи. Теорію розібрав. Всі завдання були структуровано розібрані та як слід для початку обдумані. Спочатку уважно проаналізував, що саме потрібно для виконання лабораторних завдання. Провів розумову роботу та наперед визначився з тим, скільки мені потрібно змінних, проробив алгоритмізацію коду на папері, і вже аж потім, після аналізу, створив тестові набори. Тобто, те, що я очікував би отримати від власного коду - очікуваний результат. Вибрав вхідні та вихідні дані, зробив обрахунки. В першому завданні довелось створити власну статичну бібліотеку з функцією, яка б обраховувала за деякою формулою значення змінної. Для цієї статичної бібліотеки згодом створив заголовковий файл для звертання до цієї статичної бібліотеки та записав прототип функції. Потім розробив програмним кодом рішення. Також створив тестовий драйвер, який, цілком вірно, довів справність даного модулю. В наступному завданні довелося попрацювати з створенням інформації про розробника ПЗ та введенням символа копірайту “©”. З цим не виникло ніяких проблем. Того, ж приступив до виконання наступного завдання. В наступному пункті треба було розробити функцію порівняння яка видавала б не просто логічний вираз True або False, а в значенні 1 або 0. З цим також не виникло ніяких проблем. В наступному підпункті, за допомогою розробленої статичної бібліотеки та функції, треба було обчислити значення введених змінних і відобразити її в двох різних системах числення, а саме в десятковій та шістнадцятковій системах числення. Для відображення числа в шістнядцятковій системі числення довелось використати “hex”. Всі завдання в цілому були виконані. Жодних проблем не виникло. Також навчився працювати з гітхабом. Та набув деяких вмінь реалізації метода функціональної декомпозиції задач. В цілому, від цієї лабораторної роботи отримав тільки позитивні емоції, і вона виявилась не такою важкою як на це можна було очікувати, було цікаво розібрати роботу GitHub, основні його принципи, команди для поширення та загрузки коду. Саме цей програмний засіб виявився набагато зручнішим для поширення коду, ніж інші будь які засоби поширення власного коду. Так, довелось трохи розібратись, але коли до цього звикаєш, то гіт стає ледь не найпростішим текстовим методом для поширення свого коду з іншими. Текстова версія едітора виявилась для мене в рази зручнішої та швишдої аніж її аналог з інтерфейсом. Лабораторна робота сподобалась.

ДОДАТОК А

|  |  |
| --- | --- |
| Назва тестового набору  Test Suite Description | Task\_8\_1 |
| Назва проекта / ПЗ  Name of Project / Software | BMTP-Lab08 |
| Рівень тестування  Level of Testing | системний / System Testing |
| Автор тест-сьюта  Test Suite Author |  |
| Виконавець  Implementer | Kobiz Andriy |

|  |  |  |  |
| --- | --- | --- | --- |
| Ід-р тест-кейса / Test Case ID | Дії (кроки) /  Action  (Test Steps) | Очікуваний  результат /  Expected Result | Результат тестування (пройшов/не вдалося/ заблокований) /  Test Result (passed/failed/ blocked) |
| 01 | Ввести: x = 10, y = 10, z = 10 | 888.53 | passed |
| 02 | Ввести: x = 44, y = 22, z = 11 | 5916.97 | passed |
| 03 | Ввести: x = 11, y = 33, z = 22 | 4338.99 | passed |
| 04 | Ввести: x = 33, y = 1, z = 6 | 1591.11 | passed |
| 05 | Ввести: x = 8, y = 16, z = 24 | 2286.23 | passed |
| 06 | Ввести: x = 10, y = 49, z = 13 | 5170.92 | passed |
| 07 | Ввести: x = 7, y = 32, z = 11 | 2487.97 | passed |
| 08 | Ввести: x = 90, y = 40, z = 20 | 22483.78 | passed |
| 09 | Ввести: x = 0, y = 0, z = 10 | 88.53 | passed |
| 10 | Ввести: x = -13, y = 10, z = 10 | 37.53 | passed |

|  |  |  |
| --- | --- | --- |
| ID  TS | Вхідні дані | Вихідні дані |
| 1 | 10,10,10 | 888.53 |
| 2 | 44,22,11 | 5916.97 |
| 3 | 11,33,22 | 4338.99 |
| 4 | 33,1,6 | 1591.12 |
| 5 | 8,16,24 | 2286.24 |
| 6 | 10,49,13 | 5170.93 |
| 7 | 7,32,11 | 2487.97 |
| 8 | 40,40,20 | 9983.79 |
| 9 | 0,0,10 | 88.53 |
| 10 | -13,10,10 | 37.53 |

ДОДАТОК Б

|  |  |
| --- | --- |
| Назва тестового набору  Test Suite Description | Task\_8\_2 |
| Назва проекта / ПЗ  Name of Project / Software | BMTP-Lab08 |
| Рівень тестування  Level of Testing | системний / System Testing |
| Автор тест-сьюта  Test Suite Author |  |
| Виконавець  Implementer | Kobiz Andriy |

|  |  |  |  |
| --- | --- | --- | --- |
| Ід-р тест-кейса / Test Case ID | Дії (кроки) /  Action  (Test Steps) | Очікуваний  результат /  Expected Result | Результат тестування (пройшов/не вдалося/ заблокований) /  Test Result (passed/failed/ blocked) |
| 01 | Ввести: 1 2 3 a p | 1  29  1d | passed |
| 02 | Ввести: 13 48 21 z s | 0  6707  1a33 | passed |
| 03 | Ввести: 0 -5 1 a a | 0  12  c | passed |
| 04 | Ввести: 49 11 55 o r | 1  13198  338e | passed |
| 05 | Ввести: 0 1 2 z a | 0  3  3 | passed |

ДОДАТОК В

Лістинг ModulesKobiz

=============================================================

#include <cmath>

#include <iomanip>

#include <iostream>

using namespace std;

float s\_calculation(int x,int y,int z){

float x1 = x;

float y1 = y;

float z1 = z;

float s = pow((x1+y1+z1),2) - (6.28\*sqrt(z1/3));

s = round(s\*100)/100;

return s;

}

=============================================================

ДОДАТОК Г

Лістинг TestDriver

==============================================================

#include <iostream>

#include <ModulesKobiz.h>

using namespace std;

void test\_calculation(){

float exp\_res [10] = {888.53, 5916.97, 4338.99, 1591.12, 2286.24, 5170.93, 2487.97, 9983.79, 88.53, 37.53};

float res [10];

int datain [10][3] = {{10, 10, 10},

{44, 22 ,11},

{11, 33 ,22},

{33, 1, 6},

{8, 16, 24},

{10, 49, 13},

{7, 32, 11},

{40, 40, 20},

{0, 0, 10},

{-13, 10 ,10}

};

for(int i = 0; i < 10; i++){

res[i] = s\_calculation(datain[i][0], datain[i][1], datain[i][2]);

if(res[i] == exp\_res[i])

cout << "Testcase #" << i+1 << " is Passed."<< endl;

else

cout << "Testcase #" << i+1 << " is Failed." << endl;

}

}

int main()

{

test\_calculation();

return 0;

}

==============================================================