МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

ЗВІТ

ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 9

з навчальної дисципліни

“Базові методології та технології програмування”

РЕАЛІЗАЦІЯ ПРОГРАМНИХ МОДУЛІВ РОЗГАЛУЖЕНИХ ТА ІТЕРАЦІЙНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ

ВИКОНАВ

студент академічної групи КБ-23

\_\_\_\_\_\_\_\_ Олексій КОНСТАНТИНОВ

ПЕРЕВІРИЛА

викладач кафедри кібербезпеки

та програмного забезпечення

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Ганна ДРЄЄВА

Кропивницький – 2023

**ТЕМА:** **РЕАЛІЗАЦІЯ ПРОГРАМНИХ МОДУЛІВ РОЗГАЛУЖЕНИХ ТА ІТЕРАЦІЙНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ**

**МЕТА:** Набути ґрунтовних вмінь і практичних навичок реалізації технології модульного програмування, застосування операторів С/С++ арифметичних, логічних, побітових операцій, умови, циклів та вибору під час розроблення статичних бібліотек, заголовкових файлів та програмних засобів у кросплатформовому середовищі Code::Blocks.

**ЗАВДАННЯ:**

1. Реалізувати функції розв’язування задач 9.1–9.3 як складових статичної бібліотеки libModulesKonstantynov.а (проект ModulesKonstantynov, створений під час виконання лабораторної роботи №8).

2. Реалізувати програмне забезпечення розв’язування задачі 9.4 на основі функцій статичної бібліотеки libModulesKonstantynov.а.

Задача 9.1.

Користувач вводить висоту хвилі, модуль виводить бал хвилювання моря.

1) Аналіз задачі 9.1:

Вхідні дані: висота хвилі

Вихідні дані: бал хвилювання моря

2) Алгоритм:

1. Початок.

2. Отримання значення висоти хвиль (waveHeight).

3. Якщо висота хвиль дорівнює 0, повернути 0.

4. Якщо висота хвиль знаходиться у діапазоні від 0 до 0.1, повернути 1.

5. Якщо висота хвиль знаходиться у діапазоні від 0.1 до 0.5, повернути 2.

6. Якщо висота хвиль знаходиться у діапазоні від 0.5 до 1.25, повернути 3.

7. Якщо висота хвиль знаходиться у діапазоні від 1.25 до 2.5, повернути 4.

8. Якщо висота хвиль знаходиться у діапазоні від 2.5 до 4, повернути 5.

9. Якщо висота хвиль знаходиться у діапазоні від 4 до 6, повернути 6.

10. Якщо висота хвиль знаходиться у діапазоні від 6 до 9, повернути 7.

11. Якщо висота хвиль знаходиться у діапазоні від 9 до 14, повернути 8.

12. У протилежному випадку повернути 9.

13. Кінець.

3) Лістинг модуля й тестового драйвера знаходиться у додатках 1 та 2 відповідно.

Задача 9.2.

Користувач вводить температуру за шкалою Фаренгейта, а на виході отримує температуру за шкалою Цельсія.

Вхідні дані: температура за шкалою Фаренгейта

Вихідні дані: температура за шкалою Цельсія

2)Алгоритм:

1. Початок.

2. Отримання значення температури у градусах Фаренгейта (tFar).

3. Обчислення температури у градусах Цельсія за формулою:

tCel = (tFar - 32) \* 5 / 9.

4. Повернення значення температури у градусах Цельсія (tCel).

5. Кінець.

3) Лістинг модуля й тестового драйвера знаходиться у додатках 1 та 2 відповідно.

Задача 9.3.

Користувач вводить натуральне число від 0 до 65535. Якщо біт цього числа рівний 1, програма виводить кількість двійкових нулів у ньому, інакше – суму двійкових одиниць.

1) Аналіз задачі 9.3:

Вхідні дані: натуральне число від 0 до 65535

Вихідні дані: кількість двійкових нулів або сума двійкових одиниць

2) Алгоритм:

1. Початок.

2. Отримання значення числа (number).

3. Перевірка, чи число менше або дорівнює 1 або більше за 65535. Якщо так, повернення -1.

4. Початок циклу: для кожного цілого числа (і) від 2 до квадратного кореня з числа number включно:

5. Перевірка, чи number ділиться націло на i. Якщо так, повернення -1, інакше – якщо останній біт числа 1 – обчислення кількості двійкових нулів у числі, інакше – кількості двійкових одиниць.

6. Повернення отриманого результату.

7. Кінець.

3) Лістинг модуля й тестового драйвера знаходиться у додатках 1 та 2 відповідно.

Результат виконання модульного тестування:

Error: invalid input Expect: Error passed

Input height: 0 Grade: 0 Expected grade:0 passed

Input height: 0.05 Grade: 1 Expected grade:1 passed

Input height: 0.3 Grade: 2 Expected grade:2 passed

Input height: 1.1 Grade: 3 Expected grade:3 passed

Input height: 2 Grade: 4 Expected grade:4 passed

Input height: 2.8 Grade: 5 Expected grade:5 passed

Input height: 5 Grade: 6 Expected grade:6 passed

Input height: 7 Grade: 7 Expected grade:7 passed

Input height: 10 Grade: 8 Expected grade:8 passed

Input height: 20 Grade: 9 Expected grade:9 passed

Temperature (far): -60 Temperature (cel): -51 Expected temperature: -51 passed

Temperature (far): 100 Temperature (cel): 37 Expected temperature: 37 passed

Temperature (far): 0 Temperature (cel): -17 Expected temperature: -17 passed

Temperature (far): 210 Temperature (cel): 98 Expected temperature: 98 passed

Temperature (far): -13 Temperature (cel): -25 Expected temperature: -25 passed

Error: invalid input Expect: Error passed

Error: invalid input Expect: Error passed

Error: invalid input Expect: Error passed

Input number: 2 Result: 2 Expected result : 2 passed

Input number: 3 Result: 14 Expected result : 14 passed

Input number: 5 Result: 14 Expected result : 14 passed

Input number: 7 Result: 13 Expected result : 13 passed

Висновок. У ході виконання лабораторної роботи, я реалізував функції розв’язування задач трьох як складових статичної бібліотеки, реалізував програмне забезпечення розв’язування задачі 9.4 на основі функцій раніше створеної статичної бібліотеки. Реалізував валідацію для кожної введеної змінної, використав побітові операції, працював із масивами, розгалуженнями та різними видами циклів.

Додаток 1. Вихідний код TestDriver

#include <iostream>

#include "..\..\..\lab08\prj\ModulesKonstantynov\ModulesKonstantynov.h"

using namespace std;

int main() {

float input[]{ -1,0,0.05,0.3,1.1,2,2.8,5,7,10,20 };

float expectedOutput[]{ 0,0,1,2,3,4,5,6,7,8,9 };

for (int i = 0; i < 11; i++) {

if (getGrade(input[i]) == 10) {

cout << "Error: invalid input\tExpect: Error\tpassed\n";

}

else {

cout << "Input height: " << input[i] << "\tGrade: " << getGrade(input[i]) << "\t" << "Expected grade:" << expectedOutput[i] << "\t";

if (getGrade(input[i]) == expectedOutput[i]) {

cout << "passed\n";

}

else {

cout << "failed\n";

}

}

}

int input2[]{ -60,100,0,210,-13 };

int expectedOutput2[]{ -51,37,-17,98,-25 };

for (int i = 0; i < 5; i++)

{

cout << "Temperature (far): " << input2[i] << "\tTemperature (cel): " << temperature(input2[i]) << "\tExpected temperature: " << expectedOutput2[i];

if (temperature(input2[i]) == expectedOutput2[i]) {

cout << "\tpassed\n";

}

else {

cout << "\tfailed\n";

}

}

int input3[]{ -5,0,1,2,3,5,7 };

int expectedOutput3[]{ 2,14,14,13 };

for (int i = 0; i < 7; i++) {

if (primalNum(input3[i]) == -1) {

cout << "Error: invalid input\tExpect: Error\tpassed\n";

}

else {

cout << "Input number: " << input3[i] << "\tResult: " << primalNum(input3[i]) << "\tExpected result : " << expectedOutput3[i-3];

if (primalNum(input3[i]) == expectedOutput3[i-3]) {

cout << "\tpassed\n";

}

else {

cout << "\tfailed\n";

}

}

}

system("pause");

return 0;

}

Додаток 2. Вихідний код ModulesKonstantynov

#define \_USE\_MATH\_DEFINES

#include<cmath>

#include <bitset>

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

float s = log(fabs(sin(z))) + (0.5 \* x \* x - pow(fabs((pow((y + z), 2)) - pow(x, 5)), 0.5) / (10 \* M\_PI \* pow(z, 4)));

return s;

}

int getGrade(float waveHeight) {

if (waveHeight == 0) {

return 0;

}

else if (waveHeight > 0 && waveHeight <= 0.1) {

return 1;

}

else if (waveHeight > 0.1 && waveHeight <= 0.5) {

return 2;

}

else if (waveHeight > 0.5 && waveHeight <= 1.25) {

return 3;

}

else if (waveHeight > 1.25 && waveHeight <= 2.5) {

return 4;

}

else if (waveHeight > 2.5 && waveHeight <= 4) {

return 5;

}

else if (waveHeight > 4 && waveHeight <= 6) {

return 6;

}

else if (waveHeight > 6 && waveHeight <= 9) {

return 7;

}

else if (waveHeight > 9 && waveHeight <= 14) {

return 8;

}

else if (waveHeight > 14) {

return 9;

}

else return 10;

}

int temperature(int tFar) {

int tCel = (tFar - 32) \* 5 / 9;

return tCel;

}

int primalNum(int number) {

if (number <= 1 || number > 65535) {

return -1;

}

for (int i = 2; i < number; ++i) {

if (number % i == 0) {

return -1;

}

else {

return (number & 1) ? 16 - std::bitset<16>(number).count() : std::bitset<16>(number).count();

}

}

}

Додаток 3. Результати системного тестування

|  |  |
| --- | --- |
| Назва тестового набору  Test Suite Description | TS\_lab9 |
| Назва проекта / ПЗ  Name of Project / Software | Konstantynov-task.exe |
| Рівень тестування  Level of Testing | системний / System Testing |
| Автор тест-сьюта  Test Suite Author | Олексій Константинов |
| Виконавець  Implementer | Олексій Константинов |

|  |  |  |  |
| --- | --- | --- | --- |
| Ід-р тест-кейса / Test Case ID | Дії (кроки) /  Action (Test Steps) | Очікуваний  результат /  Expected Result | Результат тестування /  Test Result |
| 1 | 1. Відкрити застосунок  2. Ввести n | Press j to begin: n  Press V or A to exit... | passed |
| 2 | 1. Відкрити застосунок  2. j | Press j to begin: j  Input high of the wave: | passed |
| 3 | 1. Відкрити застосунок  2. Ввести m  3. Ввести n  4. Ввести j | Press j to begin: m  Press V or A to exit...n  Press j to begin: j  Input high of the wave: | passed |
| 4 | 1. Відкрити застосунок  2. Ввести j  3. Ввести 0  4. Ввести 0  5. Ввести 3 | Press j to begin: j  Input high of the wave: 0  Input the temperature: 0  Input the primal number:3  Result: -0.00943949  Press V or A to exit... | passed |
| 5 | 1. Відкрити застосунок  2. Ввести j  3. Ввести 0  4. Ввести 0  5. Ввести 0 | Press j to begin: j  Input high of the wave: 0  Input the temperature: 0  Input the primal number:0  Error: invalid input  Press j to begin: | passed |
| 6 | 1. Відкрити застосунок  2. Ввести j  3. Ввести -1 | Press j to begin: j  Input high of the wave: -1  Error: invalid input  Press j to begin: | passed |
| 7 | 1. Відкрити застосунок  2. Ввести j  3. Ввести 0  4. Ввести 100  5. Ввести 4 | Input high of the wave: 0  Input the temperature: 100  Input the primal number:4  Error: invalid input  Press j to begin: | passed |
| 8 | 1. Відкрити застосунок  2. Ввести j  3. Ввести 40  4. Ввести -10  5. Ввести 13  6. Ввести v | Press j to begin: j  Input high of the wave: 40  Input the temperature: -10  Input the primal number:13  Result: 39.6326  Press V or A to exit...v  ┌────────────────────────────────┐  │Розроблено студентом групи КБ-23│  │Константиновим Олексієм │  │© Всі права захищені! │  └────────────────────────────────┘  Press any key to continue . . . | passed |
| 9 | 1. Відкрити застосунок  2. Ввести j  3. Ввести 5  4. Ввести 5  5. Ввести 5  6. Ввести V | Press j to begin: j  Input high of the wave: 5  Input the temperature: 5  Input the primal number:5  Result: 17.9905  Press V or A to exit...V  ┌────────────────────────────────┐  │Розроблено студентом групи КБ-23│  │Константиновим Олексієм │  │© Всі права захищені! │  └────────────────────────────────┘  Press any key to continue . . . | passed |
| 10 | 1. Відкрити застосунок  2. Ввести j  3. Ввести 7  4. Ввести 7  5. Ввести 7  6. Ввести A | Press j to begin: j  Input high of the wave: 7  Input the temperature: 7  Input the primal number:7  Result: 23.6328  Press V or A to exit...A  ┌────────────────────────────────┐  │Розроблено студентом групи КБ-23│  │Константиновим Олексієм │  │© Всі права захищені! │  └────────────────────────────────┘  Press any key to continue . . . | passed |