Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

Кафедра кібербезпеки та програмного забезпечення

# 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

КІ-21-1 Лісевич Д.С.

ПЕРЕВІРИВ

викладач кафедри кібербезпеки

та програмного забезпечення

\_\_\_\_\_\_\_\_\_  П.С. Усік

Кропивницький – 2022

**ЛАБОРАТОРНА РАБОТА №9**

**Тема:** Реалізація програмних модулів розгалужених та ітераційних обчислювальних процесів.

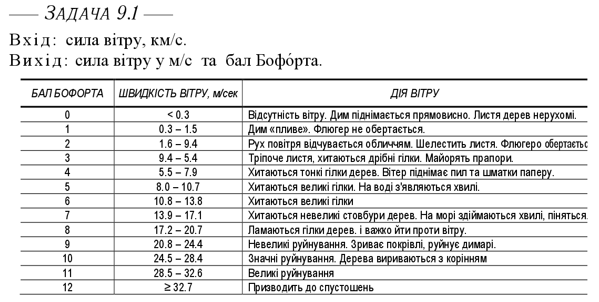
**Мета роботи:** полягає у набутті ґрунтовних вмінь і практичних навичок реалізації технології модульного програмування, застосування операторів С/С++ арифметичних, логічних, побітових операцій, умови, циклів та вибору під час розроблення статичних бібліотек, заголовкових файлів та програмних засобів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

**Варіант 16**

**Завдання:**

1. Реалізувати функції розв’язування задач 9.1–9.3 як складових статичної бібліотеки libModulesПрізвище.а (проект ModulesПрізвище, створений під час виконання лабораторної роботи №8).
2. Реалізувати програмне забезпечення розв’язування задачі 9.4 на основі функцій статичної бібліотеки libModulesПрізвище.а.

**Завдання 9.1**

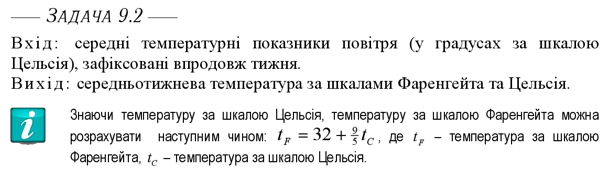
****

**Строга постанова задачі:**

**Вхідні дані:** Швидкість вітру;

**Вихідні дані:** Бал Бофорта;

**Завдання 9.2**

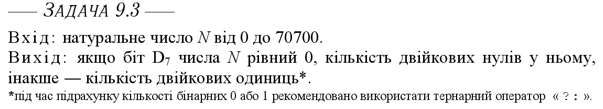
****

**Строга постанова задачі:**

**Вхідня дані:** вводиться середні температурні показники повітря зафіксовані впродовж тижня;

**Вихідні дані:** середньо тижнева температура за шкалами Фаренгейта та Цельсія;

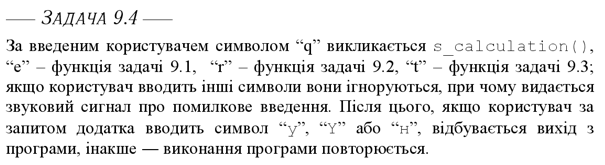
**Завдання 9.3**

****

**Строга постанова задачі:**

Вхідні та вихидні дані вказані в умові задачі.

**Завдання 9.4**

****

**Строга постанова задачі:**

**Вхідні дані:** введення користувачем символів, що вказані в умові задачі;

**Вихідні дані:** відповідна функція до символа;

**Результат запуску застосунку TestDriver:**

Test-Suite 9.1

Test Case #1

Wind Speed: 0

Expected Result: Бал Бофорта: 0

Actual Result: Бал Бофорта: 0

Test case #1 PASSED

Test Case #2

Wind Speed: 5

Expected Result: Бал Бофорта: 3

Actual Result: Бал Бофорта: 3

Test case #2 PASSED

Test Case #3

Wind Speed: 15

Expected Result: Бал Бофорта: 7

Actual Result: Бал Бофорта: 7

Test case #3 PASSED

Test Case #4

Wind Speed: 25

Expected Result: Бал Бофорта: 10

Actual Result: Бал Бофорта: 10

Test case #4 PASSED

Test Case #5

Wind Speed: 35

Expected Result: Бал Бофорта: 12

Actual Result: Бал Бофорта: 12

Test case #5 PASSED

Test-Suite 9.2.1

Test Case #1

Expected Result(Celsium): 13

Actual Result(Celsium): 13

Test case #1 PASSED

Test Case #2

Expected Result(Celsium): 5

Actual Result(Celsium): 5

Test case #2 PASSED

Test Case #3

Expected Result(Celsium): 15

Actual Result(Celsium): 15

Test case #3 PASSED

Test Case #4

Expected Result(Celsium): 20

Actual Result(Celsium): 20

Test case #4 PASSED

Test Case #5

Expected Result(Celsium): 6

Actual Result(Celsium): 6

Test case #5 PASSED

Test-Suite 9.2.2

Test Case #1

Expected Result(Fahrenheit): 55

Actual Result(Fahrenheit): 55

Test case #1 PASSED

Test Case #2

Expected Result(Fahrenheit): 41

Actual Result(Fahrenheit): 41

Test case #2 PASSED

Test Case #3

Expected Result(Fahrenheit): 59

Actual Result(Fahrenheit): 59

Test case #3 PASSED

Test Case #4

Expected Result(Fahrenheit): 68

Actual Result(Fahrenheit): 68

Test case #4 PASSED

Test Case #5

Expected Result(Fahrenheit): 42

Actual Result(Fahrenheit): 42

Test case #5 PASSED

Test task 9.3

Test-Suite #1

Expected Result: 3

Actual Result: 3

Test case #1 PASSED

Test-Suite #2

Expected Result: 26

Actual Result: 26

Test case #2 PASSED

Test-Suite #3

Expected Result: 27

Actual Result: 27

Test case #3 PASSED

Test-Suite #4

Expected Result: 6

Actual Result: 6

Test case #4 PASSED

Test-Suite #5

Expected Result: 6

Actual Result: 6

Test case #5 PASSED

**ВИСНОВОК**

Під час виконання лабораторної роботи, набув 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.

Лабораторна робота виконувалась згідно «Порядок виконання лабораторної роботи та методичні вказівки». Лабораторна робота № 9 стала другою лабораторною роботою з використанням системи контроля версій – GitHub. Не зважаючи на це, особливих проблем під час виконання завдань не виникло, а навіть навпаки було легко та цікаво ним користуватися.

Як вже було зазначено в моєму попередньому звіті з лабораторної роботи №8 з дисципліни “Базові методології та технології програмування на тему “Реалізація статичних бібліотек модулів лінійних обчислювальних процесів” мета якої 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++ 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) GitHub дає можливість розбити роботу на етапи, яка є дуже зручною, особливо сьогодні.

Одним із спостережень виконання лабораторної роботи є факт вирішення задач із попередніх лабораторних, проте з використанням модульної парадигми та більш ширшого спектру взаємодії з середовищем розробки. Керуючись строгою постановкою задачі, попередньо було виявлено та ліквідовано можливі проблеми:

* Слушною думкою було попереднє визначення кодування файлу вихідного коду (UTF-8, адже с цим кодуванням, на відміну від CP-1251 та CP-866 не виникає проблем з відображенням файлу коду на сторінці GitHub);
* Також, передбачено критичне завершення програми (або - виведення результату в консоль) у випадку отримання не коректних вхідних даних. Так в окремих модулях, хибні значення визначаються 0 (завдання 9.1: нулем визначається час роботи), або ж як -1 (в інших завданнях, окрім 9.4), що одразу свідчить про порушення умов алгоритмізації та визначення результату TestCase як Failed.

Для вирішення всіх цих завданьбула створена статична бібліотека libModulesLisevych.a в середині якої знаходиться функція s\_calculation, temp\_cel, temp\_far, bits\_number та підключені бібліотеки, зокрема: cmath, clocale та iostream. Також був створений заголовковий файл ModulesLisevych.h в середині якого знаходиться прототипи функцій s\_calculation, temp\_cel, temp\_far, bits\_number. Для того, щоб підключити створену бібліотеку було налаштовано опції проєкта Build options…:Linker; Compiler.

Під час виконання лабораторної роботи, знайшов помилку в умові задачі 9.1. Хибодрук в показниках.



Після знаходження помилки, виправив на правильний показник (9.4 – 3.4).

Завдання 9.4:

За введеним користувачем символом “q” викликається – s\_calculation(), “e” – функція задачі 9.1, “r” – функція задачі 9.2, “t” – функція задача 9.3; якщо користувач вводить інші символи, вони ігноруються, при чому видається звуковий сигнал про помилкове введення. Після цього, якщо користувач за запитом додатка вводить символ “Y”, “y”, “H” , відбувається вихід із програми, інакше – виконання програми повторюється.

Отже, ця лабораторна робота дозволила закріпити вміння та навички використання статичних бібліотек та врахування всіх аспектів пов’язаних з ними.

**ДОДАТОК А**

**Вихідний код ModulesLisevych:**

#include <iostream>

#include <clocale>

#include <cmath>

#include <string>

#include <bitset>

#include "ModulesLisevych.h"

using namespace std;

double s\_calculation(float x, float z)

{

const double PI = 3.14159;

return (0.5\*pow((abs(2\*z - pow(x, 2))/sin(x)), 3))/(sqrt(1+abs(cos(x)))+2\*PI);

}

string windSpeed(int wSpeed)

{

if(wSpeed < 0.3){

return "Бал Бофорта: 0";

}

if(wSpeed >= 0.3 && wSpeed <= 1.5){

return "Бал Бофорта: 1";

}

if(wSpeed >= 1.6 && wSpeed <= 3.4){

return "Бал Бофорта: 2";

}

if(wSpeed >= 3.5 && wSpeed <= 5.4 ){

return "Бал Бофорта: 3";

}

if(wSpeed >= 5.5 && wSpeed <= 7.9){

return "Бал Бофорта: 4";

}

if(wSpeed >= 8 && wSpeed <= 10.7){

return "Бал Бофорта: 5";

}

if(wSpeed >= 10.8 && wSpeed <= 13.8){

return "Бал Бофорта: 6";

}

if(wSpeed >= 13.9 && wSpeed <= 17.1){

return "Бал Бофорта: 7";

}

if(wSpeed >= 17.2 && wSpeed <= 20.7){

return "Бал Бофорта: 8";

}

if(wSpeed >= 20.8 && wSpeed <= 24.4){

return "Бал Бофорта: 9";

}

if(wSpeed >= 24.5 && wSpeed <= 28.4){

return "Бал Бофорта: 10";

}

if(wSpeed >= 28.5 && wSpeed <= 32.6){

return "Бал Бофорта: 11";

}

if(wSpeed >= 32.7){

return "Бал Бофорта: 12";

}

}

int temp\_cel(int day1, int day2, int day3, int day4, int day5, int day6, int day7)

{

int cel = (day1 + day2 + day3 + day4 + day5 + day6 + day7) / 7;

return cel;

}

int temp\_far(float cel)

{

int far;

far = 32 + 1.8 \* cel;

return far;

}

int bits\_number(int number)

{

bitset<32> b\_number{number};

if (b\_number[7]) {

return 32 - b\_number.count();

}

return b\_number.count();

}

**Вихідний код ModulesLisevych.h:**

#ifndef MODULESLISEVYCH\_H\_INCLUDED

#define MODULESLISEVYCH\_H\_INCLUDED

#include <iostream>

#include <string>

#include <cmath>

using namespace std;

double s\_calculation(float x, float z);

string windSpeed(int wSpeed);

int temp\_cel(int day1, int day2, int day3, int day4, int day5, int day6, int day7);

int temp\_far(float cel);

int bits\_number(int number);

#endif // MODULESLISEVYCH\_H\_INCLUDED

**ДОДАТОК Б**

**Вихідний код TestDriver:**

#include <iostream>

#include "ModulesLisevych.h"

using namespace std;

int main()

{

system("chcp 65001 & cls");

//Task 9.1

int windSpeedValue [5] = {0, 5, 15, 25, 35};

char wSpeedResult [5][255] = { "Бал Бофорта: 0", "Бал Бофорта: 3", "Бал Бофорта: 7", "Бал Бофорта: 10", "Бал Бофорта: 12"};

cout << "Test-Suite 9.1" << endl;

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

cout << "Test Case #" << i + 1 << "\n"

"Wind Speed: " << windSpeedValue[i] << "\n";

cout << "Expected Result: " << wSpeedResult[i] << endl;

cout << "Actual Result: " << windSpeed(windSpeedValue[i]) << endl;

if(windSpeed(windSpeedValue[i]) == wSpeedResult[i]){

cout << "Test case #" << i + 1 << " PASSED\n" << endl;

} else {

cout << "Test case #" << i + 1 << " FAILED\n" << endl;

}

}

//Task 9.2

int day1 [5] = {10, 5, 10, 25, 3};

int day2 [5] = {11, 7, 14, 20, 5};

int day3 [5] = {12, 9, 18, 23, 3};

int day4 [5] = {13, 7, 20, 24, 7};

int day5 [5] = {14, 5, 17, 19, 10};

int day6 [5] = {15, 3, 15, 15, 11};

int day7 [5] = {16, 5, 17, 17, 9};

int celResult [5] = {13, 5, 15, 20, 6};

float farResult [5] = {55, 41, 59, 68, 42};

cout << "Test-Suite 9.2.1" << endl;

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

cout << "Test Case #" << i + 1 << "\n";

cout << "Expected Result(Celsium): " << celResult[i] << endl;

cout << "Actual Result(Celsium): " << temp\_cel(day1[i], day2[i], day3[i], day4[i], day5[i], day6[i], day7[i]) << endl;

if(temp\_cel(day1[i], day2[i], day3[i], day4[i], day5[i], day6[i], day7[i]) == celResult[i]){

cout << "Test case #" << i + 1 << " PASSED\n" << endl;

} else {

cout << "Test case #" << i + 1 << " FAILED\n" << endl;

}

}

cout << "Test-Suite 9.2.2" << endl;

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

cout << "Test Case #" << i + 1 << "\n";

cout << "Expected Result(Fahrenheit): " << farResult[i] << endl;

cout << "Actual Result(Fahrenheit): " << temp\_far(celResult[i]) << endl;

if(temp\_far(celResult[i] == farResult[i])){

cout << "Test case #" << i + 1 << " PASSED\n" << endl;

} else {

cout << "Test case #" << i + 1 << " FAILED\n" << endl;

}

}

//Task 9.3

int number [5] = {50, 500, 5000, 50000, 70000};

int numberResult [5] = {3, 26, 27, 6, 6};

cout << "Test task 9.3" << endl;

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

cout << "Test-Suite #" << i + 1 << endl;

cout << "Expected Result: " << numberResult[i] << endl;

cout << "Actual Result: " << bits\_number(number[i]) << endl;

if(bits\_number(number[i]) == numberResult[i]){

cout << "Test case #" << i + 1 << " PASSED\n" << endl;

} else {

cout << "Test case #" << i + 1 << " FAILED\n" << endl;

}

}

}

**ДОДАТОК В**

**Вихідний код LisevychTask:**

#include <iostream>

#include <conio.h>

#include "ModulesLisevych.h"

using namespace std;

int main()

{

system("chcp 65001 & cls");

float x, z;

int wSpeed;

int day1, day2, day3, day4, day5, day6, day7;

int number;

while(true){

cout << "Введіть символ (q | e | r | t | Y | y | H): " << endl;

switch(getch()){

case 'q': cout << "Введіть значення x: " << endl;

cin >> x;

cout << "Введіть значення z: " << endl;

cin >> z;

cout << "S = " << s\_calculation(x, z) << endl;

break;

case 'e': cout << "Введіть швидкість вітру: " << endl;

cin >> wSpeed;

cout << windSpeed(wSpeed) << endl;

break;

case 'r': cout << "Введіть температуру за понеділок: " << endl;

cin >> day1;

cout << "Введіть температуру за вівторок: " << endl;

cin >> day2;

cout << "Введіть температуру за середу: " << endl;

cin >> day3;

cout << "Введіть температуру за четвер: " << endl;

cin >> day4;

cout << "Введіть температуру за п'ятницю: " << endl;

cin >> day5;

cout << "Введіть температуру за суботу: " << endl;

cin >> day6;

cout << "Введіть температуру за неділю: " << endl;

cin >> day7;

cout << temp\_cel(day1, day2, day3, day4, day5, day6, day7) << " - (Цельсій)" << endl;

cout << temp\_far(temp\_cel(day1, day2, day3, day4, day5, day6, day7)) << " - (Фаренгейт)" << endl;

break;

case 't': cout << "Введіть значення: " << endl;

cin >> number;

cout << bits\_number(number) << endl;

break;

case 'Y': return 0; break;

case 'y': return 0; break;

case 'H': return 0; break;

default: cout << "\a" << endl; continue;

}

}

}

**ДОДАТОК Г**

(Test Suite до завдання 9.4(LisevychTask))

|  |  |  |
| --- | --- | --- |
| Artifact: Test Suite | Назва тестового набору /  **Test Suite Description** | Test-Suite9.4 |
| Date: 3/28/2022 | Назва проекта/ПЗ /  **Name of project** | LisevychTask.ехе |
|  | Рівень тестування /  **Level of testing** | Системний |
|  | Автор тест-сьюта /  **Test Suite Author** | Дмитро Лісевич |
|  | Виконавець /  **Implementer** | Дмитро Лісевич |

|  |  |  |  |
| --- | --- | --- | --- |
| Ід-р тест кейса /  **Test Case ID** | Дії (кроки)  /  **Action (Test Steps)** | Очікуваний результат  /  **Expected Result** | Результат тестування  /  **Test Result** |
| TS\_01 | 1.Відкрити застосунок | Введіть символ (q | e | r | t | Y | y | H): | PASSED |
| TS\_02 | 1.Ввести q  2.Ввести 1  3.Ввести 2 | Введіть значення x:  1  Введіть значення z:  2  S = 3.01129 | PASSED |
| TS\_03 | 1.Ввести e  2.Ввести 5 | Введіть швидкість вітру:  5  Бал Бофорта: 3 | PASSED |
| TS\_04 | 1.Ввести r  2.Ввести 1  3.Ввести 2  4.Ввести 3  5.Ввести 4  6.Ввести 5  7.Ввести 6  8.Ввести 7 | Введіть температуру за понеділок:  1  Введіть температуру за вівторок:  2  Введіть температуру за середу:  3  Введіть температуру за четвер:  4  Введіть температуру за п'ятницю:  5  Введіть температуру за суботу:  6  Введіть температуру за неділю:  7  4 - (Цельсій)  39 - (Фаренгейт) | PASSED |
| TS\_05 | 1.Ввести t  2.Ввести 15 | Введіть значення:  15  4 | PASSED |
| TS\_06 | 1.Ввести Y | Застосунок закривається. | PASSED |
| TS\_07 | 1.Ввести G | Введіть символ (q | e | r | t | Y | y | H): | PASSED |

**ДОДАТОК Д**

(Test Suite до завдання 9.1)

|  |  |  |
| --- | --- | --- |
| Artifact: Test Suite | Назва тестового набору /  **Test Suite Description** | Test-Suite9.1 |
| Date: 3/28/2022 | Назва проекта/ПЗ /  **Name of project** | Lab9.1-Lisevych.ехе |
|  | Рівень тестування /  **Level of testing** | Системний |
|  | Автор тест-сьюта /  **Test Suite Author** | Дмитро Лісевич |
|  | Виконавець /  **Implementer** | Дмитро Лісевич |

|  |  |  |  |
| --- | --- | --- | --- |
| Ід-р тест кейса /  **Test Case ID** | Дії (кроки)  /  **Action (Test Steps)** | Очікуваний результат  /  **Expected Result** | Результат тестування  /  **Test Result** |
| TS\_01 | 1. Увести 0 | Бал Бофорта: 0 | PASSED |
| TS\_02 | 1. Увести 5 | Бал Бофорта: 3 | PASSED |
| TS\_03 | 1. Увести 15 | Бал Бофорта: 7 | PASSED |
| TS\_04 | 1. Увести 25 | Бал Бофорта: 10 | PASSED |
| TS\_05 | 1. Увести 35 | Бал Бофорта: 12 | PASSED |

**ДОДАТОК Е**

(Test Suite до завдання 9.2)

|  |  |  |
| --- | --- | --- |
| Artifact: Test Suite | Назва тестового набору /  **Test Suite Description** | Test-Suite9.2 |
| Date: 3/28/2022 | Назва проекта/ПЗ /  **Name of project** | Lab9.2-Lisevych.ехе |
|  | Рівень тестування /  **Level of testing** | Системний |
|  | Автор тест-сьюта /  **Test Suite Author** | Дмитро Лісевич |
|  | Виконавець /  **Implementer** | Дмитро Лісевич |

|  |  |  |  |
| --- | --- | --- | --- |
| Ід-р тест кейса /  **Test Case ID** | Дії (кроки)  /  **Action (Test Steps)** | Очікуваний результат  /  **Expected Result** | Результат тестування  /  **Test Result** |
| TS\_01 | 1. Увести значення 10;  2. Увести значення 11;  3. Увести значення 12;  4. Увести значення 13;  5. Увести значення 14;  6. Увести значення 15;  7. Увести значення 16; | Celsium = 13  Fahrenheit = 55 | PASSED |
| TS\_02 | 1. Увести значення 5;  2. Увести значення 7;  3. Увести значення 9;  4. Увести значення 7;  5. Увести значення 5;  6. Увести значення 3;  7. Увести значення 5; | Celsium = 5  Fahrenheit = 41 | PASSED |
| TS\_03 | 1. Увести значення 10;  2. Увести значення 14;  3. Увести значення 18;  4. Увести значення 20;  5. Увести значення 17;  6. Увести значення 15;  7. Увести значення 17; | Celsium = 15  Fahrenheit = 59 | PASSED |
| TS\_04 | 1. Увести значення 25;  2. Увести значення 20;  3. Увести значення 23;  4. Увести значення 24;  5. Увести значення 19;  6. Увести значення 15;  7. Увести значення 17; | Celsium = 20  Fahrenheit = 68 | PASSED |
| TS\_05 | 1. Увести значення 3;  2. Увести значення 5;  3. Увести значення 3;  4. Увести значення 7;  5. Увести значення 10;  6. Увести значення 11;  7. Увести значення 9; | Celsium = 6  Fahrenheit = 42 | PASSED |

**ДОДАТОК Є**

(Test Suite до завдання 9.3)

|  |  |  |
| --- | --- | --- |
| Artifact: Test Suite | Назва тестового набору /  **Test Suite Description** | Test-Suite9.3 |
| Date: 3/28/2022 | Назва проекта/ПЗ /  **Name of project** | Lab9.3-Lisevych.ехе |
|  | Рівень тестування /  **Level of testing** | Системний |
|  | Автор тест-сьюта /  **Test Suite Author** | Дмитро Лісевич |
|  | Виконавець /  **Implementer** | Дмитро Лісевич |

|  |  |  |  |
| --- | --- | --- | --- |
| Ід-р тест кейса /  **Test Case ID** | Дії (кроки)  /  **Action (Test Steps)** | Очікуваний результат  /  **Expected Result** | Результат тестування  /  **Test Result** |
| TS\_01 | 1. Увести 50 | Actual Result: 3 | PASSED |
| TS\_02 | 1. Увести 500 | Actual Result: 26 | PASSED |
| TS\_03 | 1. Увести 5000 | Actual Result: 27 | PASSED |
| TS\_04 | 1. Увести 50000 | Actual Result: 6 | PASSED |
| TS\_05 | 1. Увести 70000 | Actual Result: 6 | PASSED |