Міністерство освіти України

Національний технічний університет "ХПІ"

кафедра "Стратегічного управління і системного аналізу”

**Звіт**

**Лабораторна робота 3**

з дисципліни "Основи Програмування"

Виконав: студент групи КН-1223А

.Джим І.Р

Перевірив:

Івашко А.В.

Харків 2023

Зміст

[**Завдання** 3](#_Toc148688560)

[**1. Мета роботи** 3](#_Toc148688561)

[**2. Завдання для виконання** 3](#_Toc148688562)

[**Рішення** 4](#_Toc148688563)

[Завдання 1 4](#_Toc148688564)

[Завдання 2 4](#_Toc148688565)

[Завдання 3 4](#_Toc148688566)

[**Код проекту** 5](#_Toc148688567)

[**Висновок** 9](#_Toc148688568)

2

# **Завдання**

### **1. Мета роботи**

Метою лабораторної роботи є отримання практичних навичок в роботі з умовним оператором і розгалуженими алгоритмами в мові ***C***.

### **2. Завдання для виконання**

Побудувати програму, яка вводить координати точки (***x***, ***у***) і визначає, чи потрапляє точка в заштриховану область на малюнку, який відповідає виданому варіанту (див. п.6). Попадання на межу області вважати попаданням в область.

3

# **Рішення**

Цей код містить програму з трьома завданнями, які можна вибрати за допомогою меню. Давайте розглянемо кожну функцію і її роботу детальніше.

# Завдання 1

1. \*\*Task\_1()\*\*: Ця функція призначена для обчислення, чи точка з координатами `(x, y)` потрапляє в задану область. Вона вимагає введення координат `x` та `y` користувачем. Після введення вона перевіряє, чи вказана точка потрапляє в область, яку задають умови. Якщо точка задовольняє умови, виводиться "Точка потрапляє в задану область", інакше - "Точка не потрапляє в задану область".

# Завдання 2

2. \*\*Task\_2()\*\*: Ця функція вирішує квадратне рівняння вигляду `ax^2 + bx + c = 0`, де `a`, `b`, і `c` вводяться користувачем. Вона перевіряє, чи рівняння є квадратним, неповним квадратним, або не квадратним, і вирішує його залежно від умови. Результати обчислень виводяться на екран.

# Завдання 3

3. \*\*Task\_3()\*\*: Ця функція призначена для обробки кількості грибів, яку вводить користувач. В залежності від кількості грибів, виводиться правильна форма слова "гриб". Це робиться за допомогою умов, які враховують особливості закінчення слів в українській мові.

4. \*\*Menu()\*\*: Ця функція створює текстове меню для користувача, в якому він може вибрати одне з трьох завдань або завершити програму. Вибір користувача обробляється за допомогою конструкції `switch`, і викликаються відповідні функції в залежності від вибору.

5. \*\*main()\*\*: В цій функції запускається головне меню (функція `Menu()`).

4

Загалом, цей код є текстовою програмою з меню, яка дозволяє користувачу вибирати між трьома завданнями: перевірка точки на потрапляння в задану область, вирішення квадратного рівняння та обробка кількості грибів. Кожна функція виконує свою задачу, і результати виводяться на екран користувачу з відповідними відтінками кольорів для кращого виділення результатів та повідомлень про помилки.

# **Код проекту**

#include <iostream>

#include <math.h>

#include <windows.h>

#include <stdio.h>

#include <iomanip>

using namespace std;

void Task\_1 ()

{

setlocale(LC\_ALL, "Russian");

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

HANDLE h;

h = GetStdHandle(STD\_OUTPUT\_HANDLE);

SetConsoleTextAttribute(h, 14);

double x, y;

printf("Введіть координату x > ");

SetConsoleTextAttribute(h, 5);

cin >> x;

SetConsoleTextAttribute(h, 14);

printf("Введіть координату у > ");

SetConsoleTextAttribute(h, 5);

cin >> y;

printf("x=%6.3lf; y=%6.3lf\n", x, y);

SetConsoleTextAttribute(h, 14);

if ((y >= x) && (y >= -x) && (y <= 1) && (y >= -1))

printf("Точка потрапляє в задану область\n");

else {

if ((y <= 1) && (y >= -1) && (y <= x) && (y <= -x))

printf("Точка потрапляє в задану область\n");

else

{

printf("Точка не потрапляє в задану область\n");

}

}

}

void Task\_2() {

HANDLE h = GetStdHandle(STD\_OUTPUT\_HANDLE);

double a, b, c;

printf("Введіть значення a, b, c для рівняння вигляду ax^2 + bx + c = 0> ");

cin >> a >> b >> c;

printf("Аналз рівняння %lf\*x^2 %+lfx %+lf = 0\n", a, b, c);

if (a == 0) {

SetConsoleTextAttribute(h, 4);

printf("Рівняння не є квадратним!\n");

if (b == 0) {

printf("Рівняння не має дійсних коренів!\n");

return;

5

}

double x = -c / b;

cout << "Відповідь " << x << endl;

return;

}

else if (b == 0) {

SetConsoleTextAttribute(h, 4);

printf("Рівняння є неповним квадратним рівнянням!\n");

if (c > 0) {

printf("Рівняння не має дійсних коренів!\n");

SetConsoleTextAttribute(h, 15);

return;

}

else {

SetConsoleTextAttribute(h, 3);

double x1 = sqrt(-c);

double x2 = -sqrt(-c);

cout << "Відповідь " << x1 << " " << x2 << endl;

return;

}

}

else if (c == 0)

{

SetConsoleTextAttribute(h, 4);

printf("Рівняння є неповним квадратним рівнянням!\n");

if (b == 0) {

SetConsoleTextAttribute(h, 3);

double x1 = 0;

double x2 = 0;

cout << "Відповідь " << x1 << " " << x2 << endl;

}

else {

double x = -b / a;

SetConsoleTextAttribute(h, 3);

cout << "Відповідь " << x << endl;

}

return;

}

double D;

D = b \* b - 4 \* a \* c;

if (D < 0) {

SetConsoleTextAttribute(h, 4);

printf("Рівняння не має дійсних коренів!\n");

SetConsoleTextAttribute(h, 15);

}

else if (D == 0) {

double y = b / (2 \* a);

SetConsoleTextAttribute(h, 15);

printf("Дискримінант рівняння дорівнює нулю\n");

cout << "Відповідь " << y << endl;

}

else {

double x1 = (-b - sqrt(D)) / (2 \* a);

double x2 = (-b + sqrt(D)) / (2 \* a);

SetConsoleTextAttribute(h, 10);

cout << "Відповідь ";

cout << x1 << " " << x2 << endl;

}

}

6

void Task\_3() {

int mushroomCount;

HANDLE h = GetStdHandle(STD\_OUTPUT\_HANDLE);

printf(">");

cin >> mushroomCount;

char wordEnding[3];

if (mushroomCount > 10 && mushroomCount < 20)

{

wordEnding[0] = 'і';

wordEnding[1] = 'в';

wordEnding[2] = '\0';

}

else if (mushroomCount % 10 == 0 || mushroomCount % 10 >= 5)

{

wordEnding[0] = 'і';

wordEnding[1] = 'в';

wordEnding[2] = '\0';

}

else if (mushroomCount % 10 == 1)

{

wordEnding[0] = '\0';

}

else if (mushroomCount % 10 > 1 && mushroomCount % 10 < 5)

{

wordEnding[0] = 'а';

wordEnding[1] = '\0';

}

printf(" У мене ");

SetConsoleTextAttribute(h, 4);

printf("%i", mushroomCount);

SetConsoleTextAttribute(h, 15);

printf(" гриб");

SetConsoleTextAttribute(h, 10);

printf("%s\n", wordEnding);

}

void Menu()

{

int gd;

do

{

setlocale(LC\_ALL, "Russian");

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

HANDLE h;

h = GetStdHandle(STD\_OUTPUT\_HANDLE);

SetConsoleTextAttribute(h, 6);

cout << "Введіть цифру 1 для запуску першої програми\n";

cout << "Введіть цифру 2 для запуску другої програми\n";

cout << "Введіть цифру 3 для запуску третьої програми\n";

cout << "Введіть цифру 0 для виходу\n";

cout << "Дія ";

SetConsoleTextAttribute(h, 3);

cin >> gd;

SetConsoleTextAttribute(h, 7);

switch (gd)

{

7

case 1:

Task\_1();

system("pause");

system("cls");

break;

case 2:

Task\_2();

system("pause");

system("cls");

break;

case 3:

Task\_3();

system("pause");

system("cls");

break;

case 0:

SetConsoleTextAttribute(h, 4);

cout << setw(10) << "Завершення...\n";

SetConsoleTextAttribute(h, 7);

system("pause");

system("cls");

break;

default:

SetConsoleTextAttribute(h, 4);

cout << "Не правильна дія\n";

SetConsoleTextAttribute(h, 7);

system("pause");

system("cls");

break;

}

} while (gd != 0);

}

int main()

{

Menu();

}

# 

8

# **Висновок**

В ході виконання лабораторної роботи у середовищі Visual Studio були отримані важливі навички програмування та роботи з інтегрованим середовищем розробки. За допомогою мови програмування, яка була використана в цій лабораторній роботі, вдалося створити програму, яка відповідає поставленим завданням.

Одним з ключових вмінь, які були розвинуті, є вміння розробляти і відлагоджувати код. Під час лабораторної роботи було необхідно виявити та виправити помилки в програмі, що сприяє покращенню розуміння процесу відлагодження.

Також, лабораторна робота дозволила ознайомитися з основними концепціями створення графічного інтерфейсу користувача, що є важливим аспектом розробки програм. Використання графічного інтерфейсу полегшило взаємодію з користувачем та зробило програму більш зручною та доступною.

Під час виконання лабораторної роботи було надано можливість вивчити та використовувати різноманітні бібліотеки та інструменти, які розширюють можливості Visual Studio. Це дозволяє покращити продуктивність та ефективність розробки.

Завдяки цій лабораторній роботі, ми також набули досвіду у спільній роботі над програмним проектом, що може бути корисним у майбутньому професійному житті.

У підсумку, виконання цієї лабораторної роботи дало нам можливість розширити свої знання та вміння в області програмування, відлагодження коду, створення графічного інтерфейсу та спільної роботи над проектами. Все це є важливими компонентами розвитку як професійного програміста.

9