#include <iostream>

#include <cmath> // Для використання математичних функцій

using namespace std;

// Прототипи функцій (оголошення функцій, що будуть використані пізніше в коді)

void task\_if19(); // Завдання 1: знайти "особливе" число (If19)

void task\_table2(); // Завдання 2: перевірка точки у фігурі (Table 2 - Geometric figures №31)

int main() {

int menu; // Змінна для вибору завдання

cout << "Task number (1 - If19, 2 - Table 2 - Geometric figures №31): ";

cin >> menu; // Введення номера завдання

// Перемикання між завданнями за допомогою оператора switch

switch (menu) {

case 1:

task\_if19(); // Виклик функції для завдання 1

break;

case 2:

task\_table2(); // Виклик функції для завдання 2

break;

default:

// Якщо введено неправильний номер завдання, вивести повідомлення

cout << "Wrong task! (Only 1,2)" << endl;

}

system("pause"); // Затримка перед завершенням програми (корисно в Windows)

return 0; // Завершення програми

}

// Завдання 1: знайти "особливе" число серед 4 введених (If19)

void task\_if19() {

int a, b, c, d; // Чотири числа для перевірки

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\* Task 1: If19 - Odd Number \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "Enter four integers (separated by spaces): ";

cin >> a >> b >> c >> d; // Введення чотирьох чисел

// Перевірка, чи ввід був коректним

if (!cin) {

cout << "Wrong input! Must be integers." << endl;

}

else {

// Перевірка, яке з чисел є "особливим" (відрізняється від інших)

if (a != b && a != c && a != d)

cout << "The odd number is at position: 1" << endl;

else if (b != a && b != c && b != d)

cout << "The odd number is at position: 2" << endl;

else if (c != a && c != b && c != d)

cout << "The odd number is at position: 3" << endl;

else if (d != a && d != b && d != c)

cout << "The odd number is at position: 4" << endl;

else

// Якщо всі числа однакові, "особливого" числа немає

cout << "No odd number found!" << endl;

}

}

// Завдання 2: перевірка належності точки до певної області (Table 2 - Geometric figures №31)

void task\_table2() {

float x, y, r; // Координати точки та радіус круга

cout << "\*\*\*\*\*\*\*\*\*\*\* Task 2: Table 2 - Geometric figures №31 \*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "Radius r: ";

cin >> r; // Введення радіуса

// Перевірка коректності введення радіуса

if (!cin || r <= 0) {

cout << "Radius must be numeric and positive!" << endl;

return; // Завершити функцію, якщо ввід некоректний

}

cout << "Point x, y: ";

cin >> x >> y; // Введення координат точки

// Перевірка коректності введення координат точки

if (!cin) {

cout << "Coordinates must be numeric!" << endl;

return; // Завершити функцію, якщо ввід некоректний

}

// Логіка перевірки належності точки до коричневої області у 1 четверті

if (x >= 0 && y >= 0 && (x \* x + y \* y <= r \* r) && (y <= x)) {

// Точка знаходиться в коричневій області, якщо:

// 1. Вона знаходиться у верхньому правому квадранті (x ≥ 0, y ≥ 0)

// 2. Відстань від початку координат не перевищує радіус (x² + y² ≤ r²)

// 3. y менше або дорівнює x (знаходиться під бісектрисою)

cout << "In brown region!" << endl;

}

// Логіка перевірки належності точки до коричневої області 2 четверті

// Точка знаходиться в коричневій області, якщо:

// 1. Вона знаходиться у верхньому правому квадранті (x ≥ 0, y ≥ 0)

float triangle\_area = (r \* r) / 2;

if (x <= 0 && y >= 0 && (x = -r) && (y = r) && x <= -r && (y >= x)) {

// (x \* x + y \* y <= r \* r)

cout << "In brown region!" << endl;

}

else {

// У всіх інших випадках точка поза коричневою областю

cout << "Out of brown region!" << endl;

}

float P, third\_side, S;

// Знаходження периметру.

third\_side = sqrt(pow(r, 2) + pow(r, 2));

cout << "3\_side" << third\_side << endl;

P = r + r + third\_side;

cout << "perimeter is " << P << endl;

S = (r \* r) / 2;

cout << "S =" << S << endl;

}