МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського

«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів

Кафедра систем управління літальних апаратів

**Лабораторна робота № 3**

з дисципліни «Алгоритмізація та програмування»

на тему "Введення-виведення даних в С ++"

ХАІ.301. 141. 319а. 19 ЛР

Виконав студент гр. \_\_\_\_\_\_319а\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*Овчинніков Станіслав\_\_\_\_\_\_\_*

(підпис, 09.11.2024) (П.І.Б.)

Перевірив

\_\_\_\_\_\_\_\_\_ к.т.н., доц. Олена  ГАВРИЛЕНКО

(підпис, дата) (П.І.Б.)

2024

МЕТА РОБОТИ

Вивчити теоретичний матеріал щодо синтаксису у мові С ++ і подання у вигляді UML діаграм активності алгоритмів з розгалуженням та реалізувати алгоритми з використанням інструкцій умовного переходу і вибору мовою C++ в середовищі Visual Studio. Також опанувати та відпрацювати навички структурування програми з функціями.

ПОСТАНОВКА ЗАДАЧІ

Завдання 1 If 16 . Вирішити завдання на алгоритми з розгалуженням. Варіанти представлено в табл.1.

Завдання 2 геом.обл 34 . Дано координати точки на площині (x, y). Визначити, чи потрапляє точка в фігуру заданого кольору (або групу фігур) і вивести відповідне повідомлення. Варіанти фігур представлено в табл.2.

Завдання 3. Обчислити площу і периметр плоскої фігури. Варіанти редставлено в табл.2.

Завдання 4. Для вибору користувачем одного з трьох зазначених вище

завдань розробити алгоритм організації меню в командному вікні з

використанням інструкції вибору.

ВИКОНАННЯ РОБОТИ

Завдання 1 If 16

Дано три змінні дійсного типу: A, B, C. Якщо їх значення впорядковані за зростанням, то подвоїти їх; в іншому випадку замінити значення кожної змінної на протилежне. Вивести нові значення змінних A, B, C.

Вихідні дані (ім’я, опис, тип): A, B, C — три вещественных числа.

Вихiднi данi (ім’я, опис, тип): Если числа упорядочены по возрастанию (A < B < C), их значения удваиваются, и выводятся новые значения. Если числа не упорядочены по возрастанию, они заменяются на противоположные значения, и выводятся новые значения.

Алгоритм вирішення

Запросить у пользователя значения переменных A, B и C.

Проверить, упорядочены ли значения по возрастанию (A < B < C).

Если да, удвоить значения каждой переменной:

A = A \* 2

B = B \* 2

C = C \* 2

Если нет, заменить значения каждой переменной на противоположные:

A = -A

B = -B

C = -C

Вывести новые значения A, B и C.

Завершить выполнение задачи.



Рисунок 1 – If 16

Лістинг коду вирішення задачі :

#include <iostream>

#include <cmath> // для функции pow

using namespace std;

// Задача if16: Удвоение или изменение знака переменных A, B, C в зависимости от условий

void task\_if16() {

float A, B, C;

cout << "\*\*\*\*\*\*\*\*\*\* Задача if16 \*\*\*\*\*\*\*\*\*\*" << endl;

cout << "Введите значения A, B и C: ";

cin >> A >> B >> C;

if (A < B && B < C) {

// Если значения упорядочены по возрастанию

A \*= 2;

B \*= 2;

C \*= 2;

cout << "Значения удвоены: A = " << A << ", B = " << B << ", C = " << C << endl;

}

else {

// Если не по возрастанию, меняем знаки

A = -A;

B = -B;

C = -C;

cout << "Значения заменены на противоположные: A = " << A << ", B = " << B << ", C = " << C << endl;

}

}



Екран роботи програми показаний на рис 2.

Завдання 2 ( Геом обл 34 )



Вхідні дані (ім’я, опис, тип, обмеження):

r — радиус окружности (положительное число).

x — координата точки по оси x.

y — координата точки по оси y.

Вихідні дані:

in\_area — логический результат, указывающий, находится ли точка в заданной области:

true, если точка в окружности или в одном из четырёх треугольников.

false, если точка не принадлежит ни одной из фигур.

"Точка находится в заданной области!" — если точка принадлежит области.

"Точка находится вне заданной области!" — если точка вне области.

Алгоритм вирішення:

Запросить у пользователя значение радиуса r.

Если r <= 0, вывести сообщение об ошибке: "Радиус должен быть положительным числом!" и завершить выполнение задачи.

Запросить у пользователя координаты точки x и y.

Проверить, принадлежит ли точка заданной области:

Проверить, принадлежит ли точка окружности: (x \* x + y \* y <= r \* r) && (x >= 0) && (y >= 0).

Проверить, принадлежит ли точка треугольнику 1 (правый верхний): (x >= 0 && y >= 0 && y <= -x + r).

Проверить, принадлежит ли точка треугольнику 2 (левый верхний): (x <= 0 && y >= 0 && y <= x + r).

Проверить, принадлежит ли точка треугольнику 3 (левый нижний): (x <= 0 && y <= 0 && y >= x - r).

Проверить, принадлежит ли точка треугольнику 4 (правый нижний): (x >= 0 && y <= 0 && y >= -x - r).

Если точка удовлетворяет хотя бы одному из этих условий, вывести сообщение: "Точка находится в заданной области!".

В противном случае вывести сообщение: "Точка находится вне заданной области!".

Завершить выполнение задачи.



Рисунок 3- код Геом обл 5.

Лістинг коду :

#include <iostream>

#include <cmath> // для математических операций

using namespace std;

void task\_34\_variant() {

float x, y, r;

cout << "\*\*\*\*\*\*\*\*\*\* Задача 34 (вариант) \*\*\*\*\*\*\*\*\*\*" << endl;

cout << "Введите радиус r: ";

cin >> r;

// Проверка корректности радиуса

if (r <= 0) {

cout << "Радиус должен быть положительным числом!" << endl;

return;

}

cout << "Введите координаты точки (x, y): ";

cin >> x >> y;

// Условия для проверки попадания точки в различные области

bool in\_circle = (x \* x + y \* y <= r \* r) && (x >= 0) && (y >= 0); // точка в правом верхнем круге

bool in\_triangle1 = (x >= 0 && y >= 0 && y <= -x + r); // правый верхний треугольник

bool in\_triangle2 = (x <= 0 && y >= 0 && y <= x + r); // левый верхний треугольник

bool in\_triangle3 = (x <= 0 && y <= 0 && y >= x - r); // левый нижний треугольник

bool in\_triangle4 = (x >= 0 && y <= 0 && y >= -x - r); // правый нижний треугольник

// Проверка принадлежности точки к области

if (in\_circle || in\_triangle1 || in\_triangle2 || in\_triangle3 || in\_triangle4) {

cout << "Точка находится в заданной области!" << endl;

} else {

cout << "Точка находится вне заданной области!" << endl;

}

}

int main() {

task\_34\_variant(); // Вызов задачи 34 (вариант)

return 0;

}



Экран роботи програми показаний на рис 4.

Завдання 4

Вирішення задачі

Завдання 4. Для вибору користувачем одного з трьох зазначених вище авдань розробити алгоритм організації меню в командному вікні з икористанням інструкції вибору.

Вхідні дані (ім’я, опис, тип, обмеження): menu — целое число, введённое пользователем для выбора задачи:

1 — выбрать задачу if16.

2 — выбрать задачу 34 (вариант).

В зависимости от выбора задачи:

Если выбрана задача if16, потребуются значения для переменных A, B и C.

Если выбрана задача 34 (вариант), потребуются значения для радиуса r и координат точки x, y.

Вихідні дані : Сообщение о выбранной задаче или ошибке:

"Значения удвоены: A = ... , B = ... , C = ..." или "Значения заменены на противоположные: A = ... , B = ... , C = ..." — для задачи if16.

"Точка находится в заданной области!" или "Точка находится вне заданной области!" — для задачи 34 (вариант).

"Неверный номер задачи! Введите 1 или 2." — если введено некорректное значение menu.

Алгоритм вирішення:

Начало.

Вывести меню с вариантами задач:

Задача 1: Проверка чисел A, B, C и их обработка (задача if16).

Задача 2: Проверка попадания точки в сложную область (задача 34 (вариант)).

Запросить у пользователя ввод номера задачи (menu).

Проверить значение menu:

Если menu равно 1, выполнить алгоритм задачи if16.

Если menu равно 2, выполнить алгоритм задачи 34 (вариант).

Если menu не равно ни 1, ни 2, вывести сообщение об ошибке: "Неверный номер задачи! Введите 1 или 2." и вернуться к шагу 3.

Завершить выполнение программы после выполнения выбранной задачи.

Экран роботи програми показаний на рис 5.

Лістинг коду :

#include <iostream>

#include <cmath> // для математических операций

using namespace std;

// Задача if16: Удвоение или изменение знака переменных A, B, C в зависимости от условий

void task\_if16() {

float A, B, C;

cout << "\*\*\*\*\*\*\*\*\*\* Задача if16 \*\*\*\*\*\*\*\*\*\*" << endl;

cout << "Введите значения A, B и C: ";

cin >> A >> B >> C;

if (A < B && B < C) {

// Если значения упорядочены по возрастанию

A \*= 2;

B \*= 2;

C \*= 2;

cout << "Значения удвоены: A = " << A << ", B = " << B << ", C = " << C << endl;

} else {

// Если не по возрастанию, меняем знаки

A = -A;

B = -B;

C = -C;

cout << "Значения заменены на противоположные: A = " << A << ", B = " << B << ", C = " << C << endl;

}

}

// Задача 34 (вариант): Проверка попадания точки в сложную область (окружность и треугольники)

void task\_34\_variant() {

float x, y, r;

cout << "\*\*\*\*\*\*\*\*\*\* Задача 34 (вариант) \*\*\*\*\*\*\*\*\*\*" << endl;

cout << "Введите радиус r: ";

cin >> r;

// Проверка корректности радиуса

if (r <= 0) {

cout << "Радиус должен быть положительным числом!" << endl;

return;

}

cout << "Введите координаты точки (x, y): ";

cin >> x >> y;

// Условия для проверки попадания точки в различные области

bool in\_circle = (x \* x + y \* y <= r \* r) && (x >= 0) && (y >= 0); // точка в правом верхнем круге

bool in\_triangle1 = (x >= 0 && y >= 0 && y <= -x + r); // правый верхний треугольник

bool in\_triangle2 = (x <= 0 && y >= 0 && y <= x + r); // левый верхний треугольник

bool in\_triangle3 = (x <= 0 && y <= 0 && y >= x - r); // левый нижний треугольник

bool in\_triangle4 = (x >= 0 && y <= 0 && y >= -x - r); // правый нижний треугольник

// Проверка принадлежности точки к области

if (in\_circle || in\_triangle1 || in\_triangle2 || in\_triangle3 || in\_triangle4) {

cout << "Точка находится в заданной области!" << endl;

} else {

cout << "Точка находится вне заданной области!" << endl;

}

}

int main() {

int menu;

cout << "Введите номер задачи (1 для if16, 2 для 34 варианта): ";

cin >> menu;

switch (menu) {

case 1:

task\_if16(); // Вызов задачи if16

break;

case 2:

task\_34\_variant(); // Вызов задачи 34 (вариант)

break;

default:

cout << "Неверный номер задачи! Введите 1 или 2." << endl;

break;

}

return 0;

}

Алгоритм вирішення:



Экран роботи програми показаний на рис 6.

ВИСНОВКИ

Программа успешно реализует меню для выбора между двумя задачами, что позволяет пользователю интуитивно выбрать и выполнить нужную операцию. Проверка корректности ввода и модульная структура кода делают программу надежной и легко расширяемой.