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\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*Єгор* Грицан*\_\_\_\_\_\_\_*

(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 10 . Вирішити завдання на алгоритми з розгалуженням. Варіанти представлено в табл.1.

Завдання 2 геом.обл 5 . 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 10

Дано дві змінні цілого типу: A і B. Якщо їх значення не рівні, то присвоїти кожній змінній суму цих значень, а якщо рівні, то присвоїти змінним нульові значення. Вивести нові значення змінних A і B.

Вихідні дані (ім’я, опис, тип):num: преобразованное значение числа (целое), целое число

Вихiднi данi (ім’я, опис, тип): преобразованное значение числа. Если введенное число положительное, здесь будет его отрицательное значение. Если число неположительное, значение не изменяется, целое число.

message — сообщение о результате. Может быть одним из следующих:

"После преобразования: [converted\_num]", если число было положительным и преобразовано в отрицательное.

"Число не положительное!", если введенное число уже неположительное.

"Ошибка: введено не целое число!", если введено некорректное значение (не целое число), строка.

Алгоритм вирішення

Начало

Запросить у пользователя ввод целого числа num.

Проверить, корректен ли ввод:

Если ввод корректен (число является целым), перейти к следующему шагу.

Если ввод некорректен (введено не число), вывести сообщение об ошибке: "Ошибка: введено не целое число!" и завершить работу функции.

Проверить значение числа num:

Если число num положительное:

Преобразовать его в отрицательное (num = -num).

Вывести результат преобразования.

Если число num не положительное (отрицательное или ноль):

Вывести сообщение: "Число не положительное!".

Конец



Рисунок 1 – If 10

Лістинг коду вирішення задачі :

#include <iostream>

#include <limits> // Для numeric\_limits

using namespace std;

void task\_if1() {

int num;

cout << "\*\*\*\*\*\*\*\*\*\* Задача 1 \*\*\*\*\*\*\*\*\*\*" << endl;

cout << "Введите целое число: ";

cin >> num;

// Проверка корректности ввода

if (cin) {

if (num > 0) {

// Преобразование положительного числа в отрицательное

num = -num;

cout << "После преобразования: " << num << endl;

} else {

// Если число неположительное (ноль или отрицательное)

cout << "Число не положительное!" << endl;

}

} else {

// Обработка ошибки, если введено не целое число

cout << "Ошибка: введено не целое число!" << endl;

cin.clear(); // Очистка состояния потока

cin.ignore(numeric\_limits<streamsize>::max(), '\n'); // Игнорирование оставшихся символов

}

}

int main() {

task\_if1(); // Вызов задачи 1 для выполнения

return 0;

}



Екран роботи програми показаний на рис 2.

Завдання 2 ( Геом обл 5 )



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

r1 — радиус внутренних кругов (красные круги).

r2 — радиус внешнего круга (желтый круг).

a — расстояние от центра внешнего круга до центров внутренних кругов.

Вихідні дані:

area\_outer\_circle — площадь внешнего круга (вычисляется по радиусу r2), вещественное число.

area\_inner\_circles — суммарная площадь двух внутренних кругов (вычисляется по радиусу r1), вещественное число.

remaining\_area — площадь оставшегося пространства внутри внешнего круга за вычетом двух внутренних кругов, вещественное число.

Алгоритм вирішення:

Начало.

Запросить у пользователя ввод координат центра окружности (a, b).

Проверить, корректен ли ввод координат (a, b):

Если ввод некорректен (введены нечисловые значения), вывести сообщение об ошибке: "Ошибка: координаты центра окружности должны быть числами!" и завершить функцию.

Запросить у пользователя ввод радиуса окружности radius.

Проверить, корректен ли ввод радиуса:

Если радиус некорректен (введено нечисловое значение или радиус <= 0), вывести сообщение об ошибке: "Ошибка: радиус должен быть

положительным числом!" и завершить функцию.

Запросить у пользователя ввод координат точки (x, y).

Проверить, корректен ли ввод координат точки (x, y):

Если ввод некорректен (введены нечисловые значения), вывести сообщение об ошибке: "Ошибка: координаты точки должны быть числами!" и завершить функцию.

Вычислить квадрат расстояния dist между центром окружности (a, b) и точкой (x, y) по формуле: dist=(x−a)^2+(y-b)^2

Сравнить квадрат расстояния dist с квадратом радиуса radius^2:

Если dist <= radius^2, то точка находится внутри или на границе окружности. Вывести сообщение: "Точка находится внутри окружности!".

Если dist > radius^2, то точка находится вне окружности. Вывести сообщение: "Точка находится вне окружности!".

Конец.



Рисунок 3- код Геом обл 5.

Лістинг коду :

#include <iostream>

#include <cmath> // Для функции pow

#include <limits> // Для numeric\_limits

using namespace std;

void task\_geom() {

float x, y, a, b;

cout << "\*\*\*\*\*\*\*\*\*\* Задача 2 \*\*\*\*\*\*\*\*\*\*" << endl;

cout << "Введите координаты центра окружности (a, b): ";

cin >> a >> b;

// Проверка корректности введённых данных для координат центра

if (!cin) {

cout << "Ошибка: координаты центра окружности должны быть числами!" << endl;

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

return;

}

float radius;

cout << "Введите радиус окружности: ";

cin >> radius;

// Проверка корректности введённых данных для радиуса

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

cout << "Ошибка: радиус должен быть положительным числом!" << endl;

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

return;

}

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

cin >> x >> y;

// Проверка корректности введённых данных для координат точки

if (!cin) {

cout << "Ошибка: координаты точки должны быть числами!" << endl;

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

return;

}

// Вычисление квадрата расстояния между точкой и центром окружности

float dist = pow(x - a, 2) + pow(y - b, 2); // Квадрат расстояния до центра

// Проверка, находится ли точка внутри или вне окружности

if (dist <= pow(radius, 2)) {

cout << "Точка находится внутри окружности!" << endl;

} else {

cout << "Точка находится вне окружности!" << endl;

}

}

int main() {

task\_geom(); // Вызов задачи 2 для выполнения

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): choice номер задачи, который вводит пользователь (целое число: 1, 2 или 3).

Вихідні дані : message — сообщение, информирующее пользователя о выборе задачи или ошибке ввода (строка). Возможные значения: "Проверка числа и его преобразование" "Проверка попадания точки в окружность" "Вычисление площади и периметра плоской фигуры" "Неверный номер задачи! Введите 1, 2 или 3." "Ошибка: введите целое число!"

Результаты выполнения выбранной задачи будут зависеть от логики, реализованной в соответствующих функциях (task\_if1, task\_geom, task\_area\_perimeter). Выходные данные для этих задач уже были описаны ранее.

Алгоритм вирішення:

Начало.

Вывести меню с вариантами задач:

Задача 1: Проверка числа и его преобразование.

Задача 2: Проверка попадания точки в окружность.

Задача 3: Вычисление площади и периметра плоской фигуры.

Запросить у пользователя ввод номера задачи (choice).

Проверить корректность ввода:

Если ввод некорректен (например, введено не целое число), вывести сообщение об ошибке: "Ошибка: введите целое число!", очистить ввод, и вернуться к шагу 2.

Выполнить команду выбора задачи (switch):

Если choice равно 1:

Вывести сообщение: "Проверка числа и его преобразование".

Вызвать функцию для выполнения задачи 1.

Если choice равно 2:

Вывести сообщение: "Проверка попадания точки в окружность".

Вызвать функцию для выполнения задачи 2.

Если choice равно 3:

Вывести сообщение: "Вычисление площади и периметра плоской фигуры".

Вызвать функцию для выполнения задачи 3.

Если choice не равно 1, 2 или 3:

Вывести сообщение об ошибке: "Неверный номер задачи! Введите 1, 2 или 3." и вернуться к шагу 2.

Завершить программу, если выбрана и выполнена одна из задач (1, 2 или 3).

Экран роботи програми показаний на рис 5.

Лістинг коду :

#include <iostream>

#include <limits> // Для numeric\_limits

using namespace std;

// Объявления функций для каждой задачи (реализация будет ниже)

void task\_if1();

void task\_geom();

void task\_area\_perimeter();

int main() {

int choice;

while (true) { // Цикл для повторного вывода меню, если ввод некорректен

cout << "Выберите задачу для выполнения:" << endl;

cout << "1 - Проверка числа и его преобразование" << endl;

cout << "2 - Проверка попадания точки в окружность" << endl;

cout << "3 - Вычисление площади и периметра плоской фигуры" << endl;

cout << "Введите номер задачи (1, 2 или 3): ";

cin >> choice;

if (!cin) {

// Проверка на корректность ввода

cout << "Ошибка: введите целое число!" << endl;

cin.clear(); // Сбрасываем флаг ошибки потока

cin.ignore(numeric\_limits<streamsize>::max(), '\n'); // Очищаем поток ввода

continue;

}

switch (choice) {

case 1:

cout << "Проверка числа и его преобразование" << endl;

task\_if1(); // Вызов функции для Задачи 1

break;

case 2:

cout << "Проверка попадания точки в окружность" << endl;

task\_geom(); // Вызов функции для Задачи 2

break;

case 3:

cout << "Вычисление площади и периметра плоской фигуры" << endl;

task\_area\_perimeter(); // Вызов функции для Задачи 3

break;

default:

cout << "Неверный номер задачи! Введите 1, 2 или 3." << endl;

continue; // Возвращаемся к началу цикла для повторного ввода

}

break; // Завершаем цикл, если ввод корректный и задача выполнена

}

return 0;

}

// Реализации функций для каждой задачи

// Задача 1: Проверка числа и его преобразование

void task\_if1() {

int num;

cout << "Введите целое число: ";

cin >> num;

if (cin) { // Проверка на корректность ввода

if (num > 0) {

num = -num;

cout << "После преобразования: " << num << endl;

} else {

cout << "Число не положительное!" << endl;

}

} else {

cout << "Ошибка: введено не целое число!" << endl;

cin.clear(); // Очистка потока ввода

cin.ignore(numeric\_limits<streamsize>::max(), '\n'); // Пропуск оставшихся символов

}

}

// Задача 2: Проверка попадания точки в окружность

void task\_geom() {

float x, y, a, b;

cout << "Введите координаты центра окружности (a, b): ";

cin >> a >> b;

if (!cin) {

cout << "Ошибка: координаты центра окружности должны быть числами!" << endl;

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

return;

}

float radius;

cout << "Введите радиус окружности: ";

cin >> radius;

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

cout << "Ошибка: радиус должен быть положительным числом!" << endl;

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

return;

}

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

cin >> x >> y;

if (!cin) {

cout << "Ошибка: координаты точки должны быть числами!" << endl;

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

return;

}

float dist = pow(x - a, 2) + pow(y - b, 2); // Квадрат расстояния до центра

if (dist <= pow(radius, 2)) {

cout << "Точка находится внутри окружности!" << endl;

} else {

cout << "Точка находится вне окружности!" << endl;

}

}

// Задача 3: Вычисление площади и периметра круга (пример для круга)

void task\_area\_perimeter() {

float radius;

cout << "Введите радиус круга: ";

cin >> radius;

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

cout << "Ошибка: радиус должен быть положительным числом!" << endl;

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

return;

}

float area = M\_PI \* radius \* radius; // Площадь круга

float perimeter = 2 \* M\_PI \* radius; // Периметр круга

cout << "Площадь круга: " << area << endl;

cout << "Периметр круга: " << perimeter << endl;

}

Алгоритм вирішення:



Экран роботи програми показаний на рис 6.

ВИСНОВКИ

Программа успешно реализует меню для выбора задач с помощью оператора switch, что позволяет пользователю интуитивно выбирать и выполнять необходимую задачу. Реализована проверка корректности ввода, что повышает надежность программы и предотвращает ошибки. Модульная структура кода делает программу гибкой и легко расширяемой.