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»

Тема: "Реалізація алгоритмів з розгалуженням мовою С ++"

ХАІ.301.141.319а. 27 ЛР

Виконав студент гр. \_\_ 319а \_\_\_\_\_\_

Руднєв Владислав

(Підпис,дата) (П.І.Б.)

Перевірив \_\_\_\_\_\_\_\_\_ к.т.н., доц.

(вчена ступінь, вчене звання)

 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. Вирішити завдання на алгоритми з розгалуженням. Варіанти

# представлено в If18.

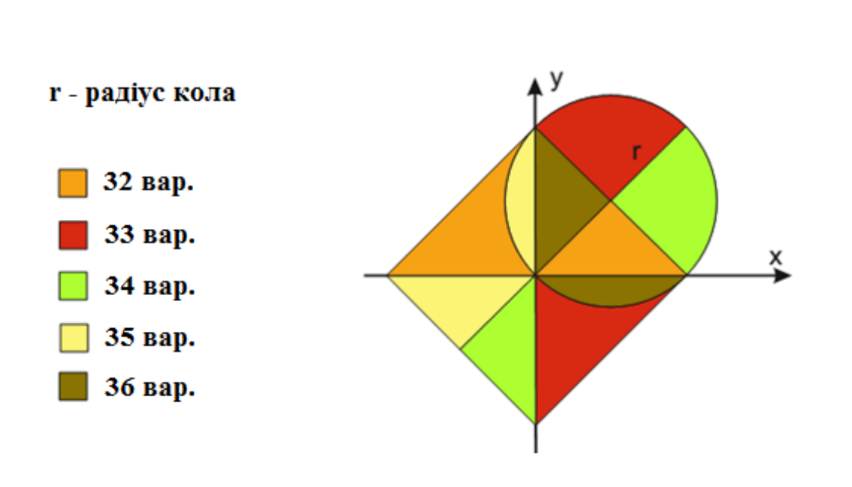
If18. Дано три цілих числа, одне з яких відмінно від двох інших, рівних між собою. Визначити порядковий номер числа, відмінного від інших.

Завдання 2. Дано координати точки на площині (x, y). Визначити, чи

потрапляє точка в фігуру заданого кольору (або групу фігур) і вивести

відповідне повідомлення.

Завдання 3. Обчислити площу і периметр плоскої фігури.

Варіант 32

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.

Вирішення задачі If18.

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

1. Ім’я змінної: a, b, c
2. Опис: число
3. Тип: int

Обмеження: a != b && a != c; b != a && b != c; c != a && c != b;

Алгоритм:

1.Введення даних: Одне число яке не дорівнює двом іншим числом.

2.Перевірка умови: Якщо числа рівні вибити помилку, якщо якесь з чисел відрізняються запустити програму.

3.Виведення результату: Вивести на екран програми колонку у якій число відрізняється.

Завдання 2.

Вирішення задачі 32

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

1. Ім’я змінної: x
2. Опис: число
3. Тип: int
4. Обмеження: лише у 1 чверті

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

1. Ім’я змінної: y
2. Опис: число
3. Тип: int
4. Обмеження: лише у 1 чверті

Алгоритм:

1. Введення даних: Ввести коордити х,у.Вивести їх та запам’ятати.
2. Перевірити чи знаходяться координати у фігурі чи за нею.
3. Вивести на екран лежать вони у фігурі чи ні.

Завдання 2, 3.

Вирішення задачі 32 .

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

Ім’я змінної: small\_triangle\_area

Опис: число

Тип: double

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

1. Ім’я змінної: large\_triangle\_area
2. Опис: число
3. Тип: double

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

1. Ім’я змінної: remaining\_area
2. Опис: число
3. Тип: double

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

1.Ім’я змінної: small\_triangle\_p

2.Опис: число

3.Тип: double

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

1.Ім’я змінної: large\_triangle\_p

2.Опис: число

3.Тип: double

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

1.Ім’я змінної: radius

2.Опис: число

3.Тип: double

Алгоритм:

1.Введення числового значення радіусу

2.Значення підставляємо у формули та обчислюємо основу та інщі площин та периметрів .

3.Виводимо площу трикутників та їх периметри завершуєм роботу програми.

Завдання 4.

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

1. Ім’я змінної: task\_if18
2. Опис: змінна
3. Тип: void

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

1. Ім’я змінної: task\_geom32
2. Опис: змінна
3. Тип: void

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

1. Ім’я змінної: menu
2. Опис: число
3. Тип: int
4. Обмеження: Тільки 1 або 2

Алгоритм:

1. Введення 1 або 2.
2. Якщо вибрано 1 запустити код task\_if18 якщо вибрано число 2 тоді запустити task\_geom32.
3. Після виконання обраного коду припинити роботу.

Лістинг коду дод. А (стр.7 - 9 )

Скрін-шоти вікна виконання програми дод. Б (сторінка 10 - 12)

ДОДАТОК А

Лістининг коду

#include <iostream>

#include <cmath>

using namespace std;

//прототипи функції

void task\_if18();

void task\_geom32();

int main() {

int menu;

cout << "Task number:";

cin >> menu;

// перемикання між завданнями

switch (menu)

{

case 1: task\_if18(); break; // Завдання 1

case 2: task\_geom32(); break; // Завдання 2

default: cout << "Wrong task! (Only 1,2)" << endl; //повідомлення про помилку

}

system("pause");

return 0;

}

//пишу код для першої задачі

void task\_geom32() {

int x;

int y;

double radius;

// Виписую дві константи щоб не було помилки при общисленні

const double PI = 3.141592;

const double half = 0.5;

// Даю запит на введення змінних

cout << "Enter the radius of the circle: ";

cin >> radius;

cout << "Enter the x , y: ";

cin >> x >> y;

//Знаходжу основу через теорему Піфагора

double base = sqrt(pow(radius ,2) + pow(radius, 2));

//Знаходжу площу маленького трикутника

double small\_triangle\_area = (base \* base) / 4;

// Знаходжу площу для кола

double circle\_area = PI \* radius \* radius;

//Знаходжу четверту частину кола

double quarter\_circle\_area = circle\_area / 4;

//Формула площи великого трикутника

double large\_triangle\_area = (base \* base) / 2;

//Фомула знаходження коричневого еліпса

double brown\_ellipse\_area = quarter\_circle\_area - small\_triangle\_area ;

//Знаходження чистої площи великого поморанчевого трикутника

double remaining\_area = large\_triangle\_area - brown\_ellipse\_area;

//Формула пириметра маленького трикутника

double small\_triangle\_p = base + radius + radius;

//Формула гіпотенузи великого трикутника

double large\_triangle\_gip = sqrt(pow(base, 2) + pow(base, 2));

//Знаходження пиреметра великого трикутника

double large\_triangle\_p = base + base + large\_triangle\_gip;

// Перевірка на додатне значення

if (brown\_ellipse\_area < 0) {

cout << "Error: Brown ellipse area is negative. Please provide more accurate data." << endl;

}

//Превірка чи знаходить точка у площині чи виходить з неї

double distance = sqrt(x \* x \* y \* y);

if (distance <= radius) {

if (distance <= quarter\_circle\_area);

}

if (x = base) {

cout << "in region " << endl;

}

else {

cout << "out region" << endl;

}

//Виведення усі площи та пириметри

cout << "Area of the small orange triangle: " << small\_triangle\_area << endl;

cout << "Area of the large orange triangle: " << large\_triangle\_area << endl;

cout << "Remaining area after subtracting yellow ellipse: " << remaining\_area << endl;

cout << "Perimetr of small orange triangle: " << small\_triangle\_p << endl;

cout << "Perimetr of large triangle : " << large\_triangle\_p << endl;

}

void task\_if18()

{

// Оголошуємо три цілочисельні змінні

int a, b, c;

// Виводимо запит на введення чисел

cout << "Enter three integers: ";

// Зчитуємо три числа з введення

cin >> a >> b >> c;

// Перевіряємо всі можливі комбінації, де одне число відрізняється від двох інших

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

cout << "Ordinal number of the perfect number: 1" << endl; // Якщо a відрізняється від b і дорівнює c

}

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

cout << "Ordinal number of the perfect number: 2" << endl; // Якщо b відрізняється від a і дорівнює c

}

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

cout << "Ordinal number of the perfect number: 3" << endl; // Якщо c відрізняється від a і дорівнює b

}

}

ДОДАТОК Б

Скрін-шоти вікна виконання програми

На рис.1 показано код програми для If18 та для задачі 32

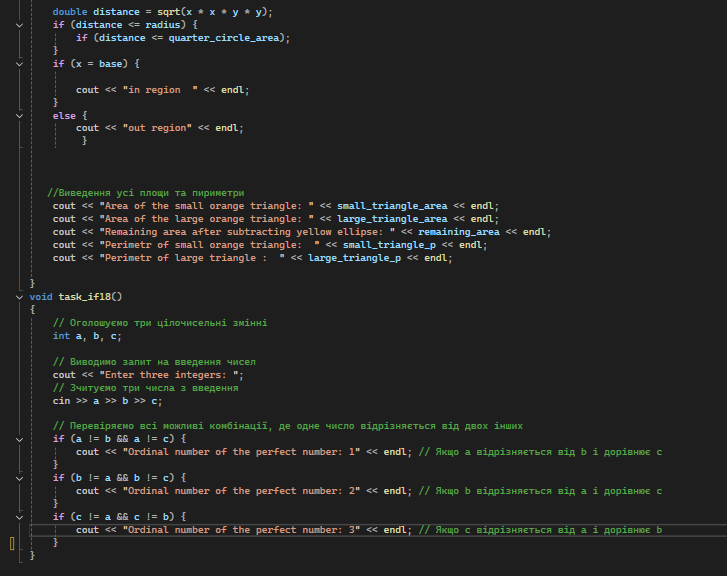
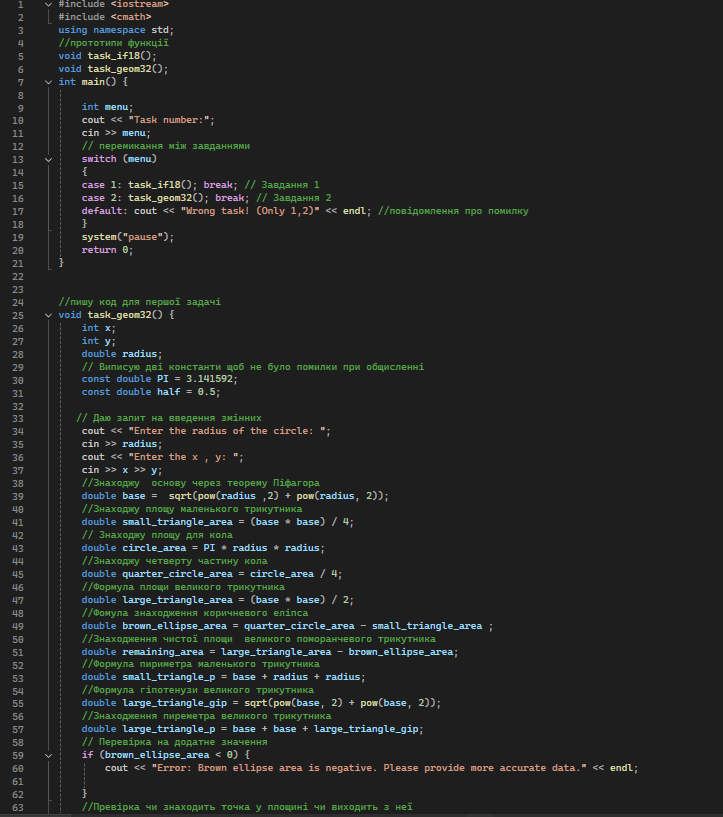


Рисунок 1 — Завдання Integer 4

На рис.2 показано код програми для Boolean27.

Рисунок 2 — Завдання Boolean 27

Рисунок 1 код програми If 18 та завдання 32 варіант 15

На рис.2 показано діаграму роботи програми if 18

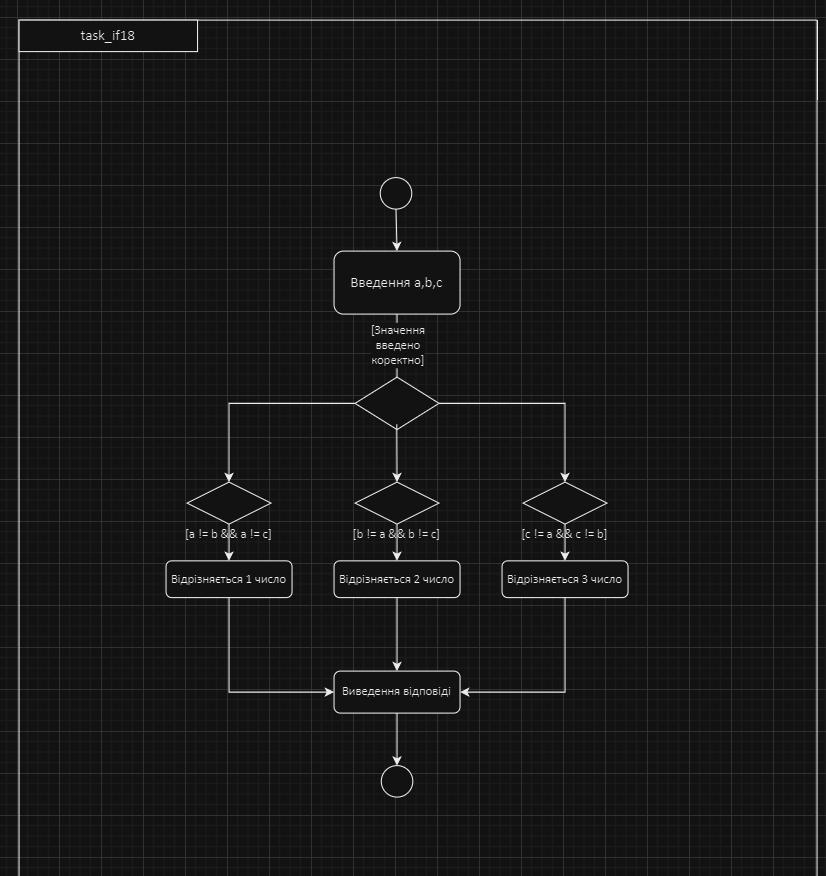


Рисунок 2 діаграма задачі if 18

На рис.3 показано діаграму роботи програми Завдання 32

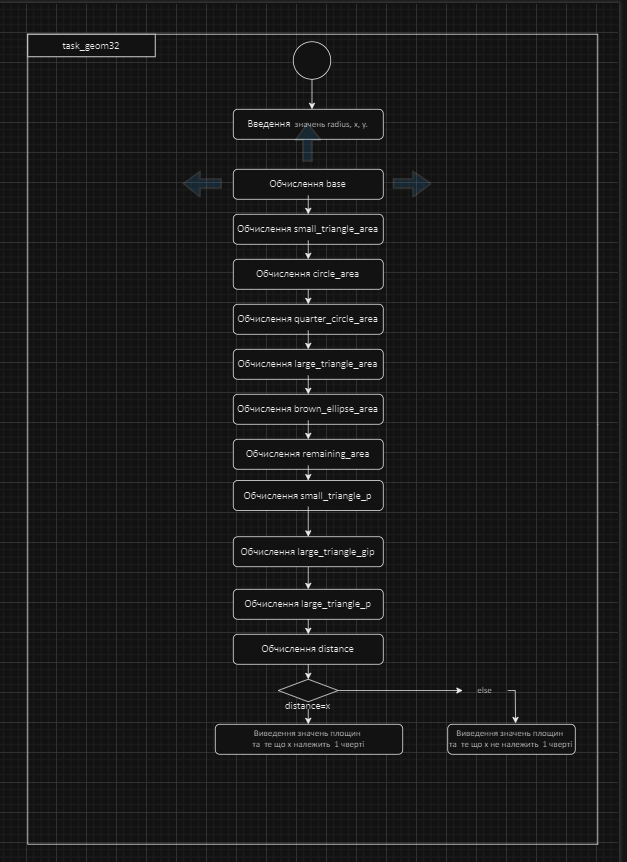


Рисунок 3 діаграма Завдання 32

Висновок

Засвоєно 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.Також покращено інші корисні навички.