Міністерство освіти і науки

Національний університет “Львівська політехніка”

**Кафедра ЕОМ**



**Звіт**

з лабораторної роботи № 3

# з дисципліни: “ Програмування, частина 2 (Об'єктно-орієнтоване програмування”

на тему: “ ЗАСОБИ РОБОТИ З ДИНАМІЧНОЮ ПАМ’ЯТТЮ. ДИНАМІЧНІ МАСИВИ ”

**виконав:**

**студент групи кі-15**

**Височанський Сергій**

**Прийняв:**

**Козак Н.Б.**

Львів 2020

**Короткі теоретичні відомості**

**Класи та об'єкти в С++**

Класи та об'єкти в С ++ є основними концепціями об'єктно-орієнтованого програмування (ООП). Об'єктно-орієнтоване програмування – розширення структурного програмування, в якому основними концепціями є поняття класів і об'єктів. Основна відмінність мови програмування С++ від С полягає в тому, що в С немає класів, а отже мова С не підтримує ООП, на відміну від С++.

**Класи в С++** – це абстракція, що описує методи і властивості ще не існуючих об'єктів. **Об'єкти** – конкретне уявлення абстракції, що має свої властивості та методи. Створені об'єкти на основі одного класу називаються **екземплярами** цього класу. Ці об'єкти можуть мати різну поведінку, властивості, але все одно будуть об'єктами одного класу. В ООП існує три основних принципи побудови класів:

1. **Інкапсуляція** – це властивість, що дозволяє об'єднати в класі і дані, і методи, які працюють з ними і приховати деталі реалізації від користувача.

2. **Успадкування** – це властивість, що дозволяє створити новий клас-нащадок на основі вже існуючого, при цьому всі характеристики класу батька присвоюються класу-нащадку.

3. **Поліморфізм** – властивість класів, що дозволяє використовувати об'єкти класів з однаковим інтерфейсом без інформації про тип і внутрішній структурі об'єкта.

**Оголошення класів в С++**

**class // ім'я класу**

**{**

**private:**

     // Список властивостей і методів для використання всередині класу

**public:**

     // Список методів доступних іншим функціям та об'єктам програми

**protected:**

     // список засобів, доступних при спадкуванні

**};**

**Завдання:**

Клас CCircle (Коло). Клас зберігає декартові координати центра кола, а також значення точки на колі. Повинні бути передбачені методи, що обчислюють довжину кола, площу та діаметр, вивід на екран поточного стану об‘єкта.

**Виконання завдання:**

#include <math.h>

class Circle

{

private:

double r;

const double pi = 3.1415926535897932384626433832795;

public:

Circle(int \*\*arr);

~Circle(){};

double lenght();

double diameter();

double square();

};

Circle::Circle(int \*\*arr) { r = sqrt(powl((arr[1][0] - arr[0][0]), 2) + powl((arr[1][1] - arr[0][1]), 2) + powl((arr[1][2] - arr[0][2]), 2)); }

double Circle::diameter() {

return (2 \* r); }

double Circle::lenght() {

return (2 \* pi \* r); }

double Circle::square() {

return (pi \* r \* r); }

#include <iostream>

using namespace std;

int main() {

setlocale(LC\_ALL, "Ukrainian");

int \*\*arr = new int \*[2];

for (int i = 0; i < 2; i++)

{

arr[i] = new int[3];

if (i == 0)

cout << "Введiть координати x, y, z центру кола:" << endl;

else if (i == 1)

cout << "Введiть координати x, y, z точки на колi:"; cin>> arr[i][0] >> arr[i][1] >> arr[i][2];

}

Circle circle(arr);

cout << "Довжина кола: " << circle.lenght() << endl;

cout << "Площа круга: " << circle.square() << endl;

cout << "Дiаметр: " << circle.diameter() << endl;

for (int i = 0; i < 2; i++)

delete[] arr[i];

delete[] arr;

}

Висновок:на даній лабораторній роботі я ознайомився з поняттям класу та обєкту в с++ а також на прикладі перевірив застосування.