Міністерство освіти і науки

Національний університет “Львівська політехніка”

**Кафедра ЕОМ**



**Звіт**

з лабораторної роботи № 4

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

на тему: “ КЛАСИ ТА ОБ’ЄКТИ ”

Виконав: ст. гр. КІ-15 Гвоздь Ю. А.

Прийняв: Козак Н. Б.

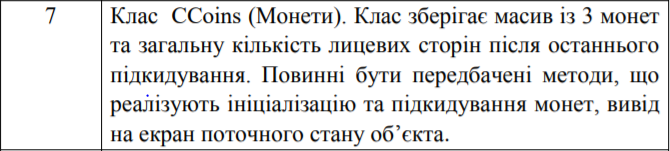
Львів – 2020

Мета: познайомитися із класами та об‘єктами.

ТЕОРЕТИЧНІ ВІДОМОСТІ

Основна відмінність будь-якої об‘єктно-орієнтованої мови програмування від інших не об‘єктно-орієнтованих мов програмування є можливість створення нових визначених користувачем типів, що називаються класами. Клас – це визначений користувачем тип з даними-елементами (властивостями) та функціями-елементами (методами), що являються членами класу. Він зазвичай описує певну абстракцію якоїсь сутності реального світу з її властивостями і можливими діями. Змінна типу клас називається об‘єктом. Об‘єкт – це вже не абстракція, а елемент реального світу, який може володіти певними характеристиками, які задаються властивостями в класі, та діяти згідно операцій заданих методами у класі. Члени класу (властивості і методи), оголошені після ключового слова publіc стають відкритими членами. Це означає, що вони доступні: • усім іншим членам класу; • дружнім конструкціям (класам, методам, функціям); • членам похідних класів; • з-під об‘єктів класу після створення об‘єктів; після створення об‘єкту класу до його членів можна звертатися лише якщо вони є загальнодоступними. Члени класу, оголошені після ключового слова prіvate, стають закритими членами. Вони доступні: • для інших членів того ж класу; • друзям класу. Якщо в класі не оголошено секцій, але визначено методи чи властивості, то вони вважатимуться такими, що оголошені у секції prіvate. Для модифікації властивостей, що оголошені в секції private створюються спеціальні методи, які здійснюють цю модифікацію. Назви таких методів прийнято починати зі слова set. На жаргоні такі методи інколи називають сетерами. Для читання властивостей, що оголошені в секції private створюються спеціальні методи, які безпечно повертають значення цих властивостей. Назви таких методів прийнято починати зі слова get. На жаргоні такі методи інколи називають ґетерами. Члени класу, оголошені після ключового слова protected, стають захищеними членами. Вони доступні: • для інших членів того ж класу; • членам похідних класів; • друзям класу. Оголошення класу містить оголошення даних-елементів (властивостей) та функцій-елементів (методів) класу. Одна з фундаментальних основ ООП передбачає інкапсуляцію даних, тобто дані мають бути недоступними ззовні, а лише за посередництвом загальнодоступних методів класу. Оголошення методів має вигляд оголошення прототипу функції в середині однієї з секцій класу, зазвичай у секції public. Якщо методи є службовими і не мають бути доступні ззовні, тоді їх розміщують в секції private або protected. Сама ж реалізація методу може знаходитись як усередині класу (тоді оголошення методу в класі має вигляд оголошення функції з тілом), так і поза оголошенням класу (тоді у класі оголошується прототип функції, а її тіло визначається за межами класу). Але один з найфундаментальніших принципів розробки програмного забезпечення мовою С++ полягає у розмежуванні інтерфейсу класу від його реалізації. Тому при побудові програми мовою C++ кожне оголошення класу зазвичай розміщують у заголовочний файл \*.h назва якого співпадає з назвою класу, а реалізацію методів цього класу – у файл \*.cpp із тим ж іменем, що й \*.h файл. Заголовочні файли включаються (за допомогою директиви #іnclude) у кожен файл, у якому використовується клас, а файли з вихідними кодами компілюються і компонуються з файлом, що містить головну програму (main-функцію). При визначенні методу класу за межами класу необхідно зв'язати ім'я відповідного методу з ім'ям класу. Зв'язок встановлюється шляхом написання імені класу, оператора розширення області видимості (::) та імені методу. Оскільки різні класи можуть мати елементи з однаковими іменами, то оператор розширення області видимості дозволяє однозначно ідентифікувати методи конкретного класу. Областю видимості властивостей та методів класу є клас, тобто все що оголошене в середині класу є видиме і доступне за іменем елементів в середині класу без додаткових маніпуляцій.

**Завдання:**



**Виконання завдання:**

Код програми:

#include <iostream>

#include <string>

#include <ctime>

using namespace std;

class CCoins {

private:

const char\* sface = "Лицева сторона";

const char\* sback = "Тильна сторона";

int coins[3] = { 0,0,0 };

int quantity = 0;

public:

void toss\_up()

{

//quantity = 0;

srand((int)time(0));

for (int i = 0; i < 3; i++) {

coins[i] = rand() % 2 + 1;

if (coins[i] == 1) {

quantity++;

} //Якщо coins[i]==1, тоді це лицева сторона,інакше тильна.

}

}

void info()

{

for (int i = 0; i < 3; i++) {

if (coins[i] == 1)

cout << sface << endl;

else

cout << sback << endl;

}

cout << "Кiлькiсть монет що є лицевими: " << quantity << endl;

cout << " " << endl;

}

void reset()

{

quantity = 0;

}

protected:

};

int main()

{

setlocale(LC\_ALL, "ukr");

int func = 1;

CCoins num1;

while (func != 4) {

cout << "Що вам потрiбно?" << endl;

cout << "Пiдкинути монети.(натиснiть 1)" << endl;

cout << "Поточний стан?(натиснiть 2)" << endl;

cout << "Скинути кiлькiсть лицевих монет.(натиснiть 3)" << endl;

cout << "Зупинити програму.(натиснiть 4)" << endl;

cout <<" "<< endl;

cin >> func;

if (func == 1) {

num1.toss\_up();

}

if (func == 2) {

num1.info();

}

if (func == 3) {

num1.reset();

cout << "Пiдкинути монети.(натиснiть 1)" << endl;

cout << "Зупинити програму.(натиснiть 4)" << endl;

cout << " " << endl;

cin >> func;

if (func == 1) {

num1.toss\_up();

}

if (func == 2) {

num1.info();

}

}

//cin >> func;

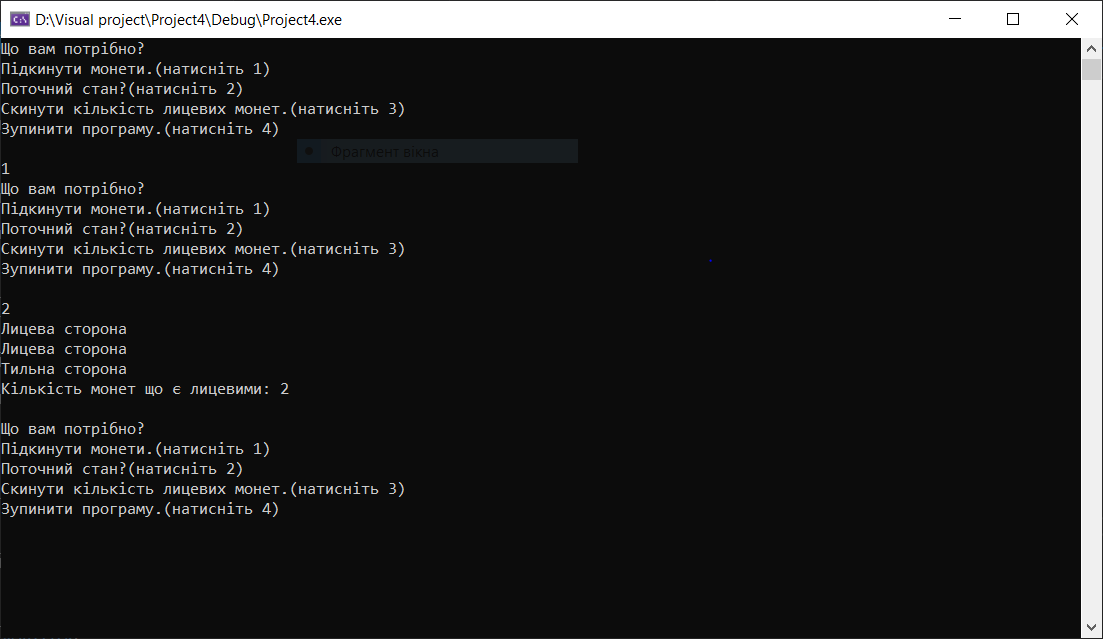
}

cout << "Програма завершила працювати.";

return 0;

}

Результат роботи програми:



Висновок: на цій лабораторній роботі я познайомився із класами та об‘єктами.