Міністерство освіти і науки України
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій

Звіт
про виконання лабораторної роботи №4
«Успадкування та поліморфізм»

Виконав:
студент групи ФЕП-11с
Качмар Д. Б

Викладач:
Кушнір Олексій
Олександрович

Львів-2025

Мета роботи: Навчитись використовувати функціонал мови C++ для створення складних гіллястих ієрархій класів.

Виконання роботи

Посилання на GitHub: https://github.com/D-Kachm/OOP/tree/main/Lab4

main.cpp > ...

```cpp
#include <iostream>
using namespace std;

class Instrument {
    string material;
    int volume;
    string sound;

public:
    int SetMaterial(string m) {
        material = m;
        return 1;
    }

    int SetVolume(int v) {
        volume = v;
        return 1;
    }

    int SetSound(string s) {
        sound = s;
        return 1;
    }

    int Play() {
        cout << "something is playing..." << endl;
        return 1;
    }

protected:
    int Describe() {
        cout << "Material: " << material << ", Volume: " << volume << ", Sound: " << sound << endl;
        return 1;
    }
};

class Guitar : public Instrument {
public:
    Guitar() {
        SetMaterial("wood");
        SetVolume(70);
        SetSound("strumming");
        Describe();
        Play();
    }

    int Play() {
        cout << "Strum-strum!" << endl;
        return 1;
    }
};

class Piano : public Instrument {
public:
    Piano() {
        SetMaterial("wood and metal");
        SetVolume(80);
        SetSound("classical chord");
        Describe();
        Play();
    }
```

```
public:
    Piano() {
    }

    int Play() {
        cout << "Plink-plonk!" << endl;
        return 1;
    }
};

class ElectricGuitar : public Guitar {
protected:
    string brand;

public:
    int SetBrand(string b) {
        brand = b;
        return 1;
    }

    ElectricGuitar(string b) {
        SetBrand(b);
        cout << "Guitar brand: " << brand << endl;
    }

    ElectricGuitar() {}
};

class Player : public ElectricGuitar {
    string name;
    string style;

public:
    int SetName(string n) {
        name = n;
        return 1;
    }

    int SetStyle(string s) {
        style = s;
        return 1;
    }

    Player(string b, string n, string s) {
        SetBrand(b);
        SetName(n);
        SetStyle(s);
        cout << name << " plays in " << style << " style on a " << brand << " guitar." << endl;
    }
};

int main() {
    Player* rockstar = new Player("Yamaha", "Andrew", "rock");
}
```

Результати

```
PS D:\University\OOP\Lab4> ./main.exe
Material: wood, Volume: 70, Sound: strumming
Strum-strum!
Andrew plays in rock style on a Yamaha guitar.
PS D:\University\OOP\Lab4>
```

Висновок: Я навчився створювати просту ієрархію класів у C++, що демонструє принципи успадкування та поліморфізму. У цьому прикладі я реалізував базовий клас Instrument та кілька похідних класів, таких як Guitar, Piano, ElectricGuitar, і Player, що перевантажують і перевизначають методи для демонстрації різних типів інструментів і їхніх характеристик. Я також використав методи для

налаштування властивостей класів та відображення результатів у консолі. Цей підхід допоміг краще зрозуміти концепції об'єктно-орієнтованого програмування, такі як поліморфізм, успадкування та перевантаження методів.