

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ТЕХНІЧНИЙ КОЛЕДЖ ТНТУ ІМЕНІ ІВАНА ПУЛЮЯ  
ВІДДІЛЕННЯ ЕЛЕКТРОННИХ АПАРАТІВ  
Циклова комісія програмних систем і комплексів

## **ЗВІТ**

про виконання лабораторних робіт з дисципліни:  
**«ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ»**

Студента 3 курсу групи КН-321  
спеціальності 122 «Комп'ютерні науки»

Попович О.В.  
(прізвище та ініціали)

Перевірів: Р.О. Слободян  
(підпис)

Тернопіль – 2020

# Лабораторна робота №5

**Тема:** Віртуальні функції та поліморфізм.

**Мета:** Практично ознайомитись з поняттям поліморфізму, його застосуванням та вивчити механізм його реалізації за допомогою віртуальних функцій

## Завдання 1

**Завдання 1.** Нехай є видавнича компанія, яка описана в завданні 1 попередньої лабораторної роботи, яка продає і книги, і аудіо версії друкованої продукції. Як і в тому завданні, створіть клас `publication`, який зберігає назву (фактично, рядок) і ціну (тип `float`) публікації. Створіть два похідних класи: `book`, який містить інформацію про кількість сторінок у книзі (типу `int`), і `tape`, який містить час запису аудіокниги у хвилинах (тип `float`). Кожен з класів повинен мати віртуальний метод `getdata()`, який буде запитувати інформацію у користувача, і віртуальний метод `putdata()` для виведення даних на екран.

Напишіть функцію `main()`, в якій створіть масив вказівників на клас `publication`:

```
publication* arr[4];
```

У циклі `while()` запитуйте у користувача, який об'єкт потрібно створити (використовуйте `new` для створення нового об'єкта `book` або `tape`). Після чого за допомогою методу `getdata()` в атрибуті об'єктів вносити дані відповідно до типу об'єкта. Коли користувач закінчить введення вихідних даних, виведіть результат для всіх введених книг і касет, використовуючи цикл `for` і єдиний вираз:

```
arr[i]->putdata();
```

для виведення даних про кожен об'єкт з масиву.

# КОД ПРОГРАМИ

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

class publication
{
protected:
    string name;
    float price;
public:
    virtual void getData() = 0;
    virtual void putdata() = 0;
};

class book : public publication
{
private:
    int numberOfPages;
public:
    book() {}
    book(string name, float price, int numberOfPages)
    {
        this->name = name;
        this->price = price;
        this->numberOfPages = numberOfPages;
    }
    void getData() override
    {
        cout << "Enter a name, price and number of pages for book" << endl;
        cin >> name >> price >> numberOfPages;
    }
    void putdata() override
    {
        cout << name << " " << price << " " << numberOfPages << " " << endl;
    }
};

class type : public publication
{
private:
    float recordingTime;
public:
    type() {}
    type(string name, float price, int numberOfPages)
    {
        this->name = name;
        this->price = price;
        this->recordingTime = recordingTime;
    }
    void getData() override
    {
        cout << "Enter a name, price and recording time for audio" << endl;
        cin >> name >> price >> recordingTime;
    }
}
```

```

        void putdata() override
        {
            cout << name << " " << price << " " << recordingTime << " " << endl;
        }
};

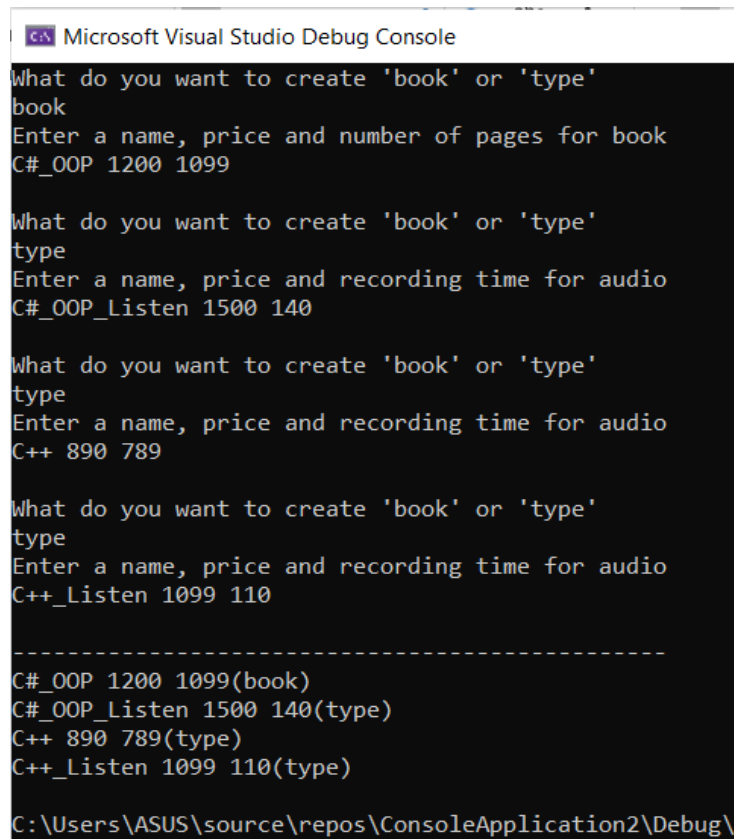
int main()
{
    const int size = 2;
    publication* pubarr[size];

    pubarr[0] = new book;
    pubarr[1] = new type;

    pubarr[0]->getData();
    pubarr[1]->getData();
    cout << "-----" << endl;
    for (int i = 0; i < size; i++)
    {
        pubarr[i]->putdata();
    }
    return 0;
}

```

## РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ



Microsoft Visual Studio Debug Console

```

What do you want to create 'book' or 'type'
book
Enter a name, price and number of pages for book
C#_OOP 1200 1099

What do you want to create 'book' or 'type'
type
Enter a name, price and recording time for audio
C#_OOP_Listen 1500 140

What do you want to create 'book' or 'type'
type
Enter a name, price and recording time for audio
C++ 890 789

What do you want to create 'book' or 'type'
type
Enter a name, price and recording time for audio
C++_Listen 1099 110

-----
C#_OOP 1200 1099(book)
C#_OOP_Listen 1500 140(type)
C++ 890 789(type)
C++_Listen 1099 110(type)

C:\Users\ASUS\source\repos\ConsoleApplication2\Debug\

```

# Завдання 2

**Завдання 2.** Взявши за основу програму із завдання 1, додайте до класів `book` і `tape` метод `isOveersize()`, який повертає значення типу `bool`. Припустимо, книга, в якій більше 800 сторінок, або аудіо запис, з часом програвання якого більше 90 хвилин, будуть вважатися об'єктами з перевищенням розміру. До цієї функції можна звертатися з `main()`, а результат її роботи виводити у вигляді рядка «Перевищення розміру!» для відповідних книг і касет. Об'єкти класів `book` і `tape` повинні зберігатися в масиві типу `publication*`.

## КОД ПРОГРАМИ

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

class publication
{
protected:
    string name;
    float price;
public:
    virtual void getData() = 0;
    virtual void putdata() = 0;
    virtual bool isOveersize() = 0;
};

class book : public publication
{
private:
    int numberOfPages;
public:
    book() {}
    book(string name, float price, int numberOfPages)
    {
        this->name = name;
        this->price = price;
        this->numberOfPages = numberOfPages;
    }
    void getData() override
    {
        cout << "Enter a name, price and number of pages for book" << endl;
```

```

        cin >> name >> price >> numberOfPages;
    }
    void putdata() override
    {
        cout << name << " " << price << " " << numberOfPages << "(book)" << endl;
    }
    bool isOveersize()override
    {
        if (numberOfPages > 800)
            return true;
        else
            return false;
    }
};

class type : public publication
{
private:
    float recordingTime;
public:
    type() {}
    type(string name, float price, int numberOfPages)
    {
        this->name = name;
        this->price = price;
        this->recordingTime = recordingTime;
    }
    void getData() override
    {
        cout << "Enter a name, price and recording time for audio" << endl;
        cin >> name >> price >> recordingTime;
    }
    void putdata() override
    {
        cout << name << " " << price << " " << recordingTime << "(type)" << endl;
    }
    bool isOveersize() override
    {
        if (recordingTime > 90)
            return true;
        else
            return false;
    }
};

int main()
{
    const int size = 4;
    publication* pubarr[size];

    int i = 0;
    while (i != size) {
        cout << "What do you want to create 'book' or 'type'" << endl;
        string choice = "";
        cin >> choice;
        if (choice == "book")
        {
            pubarr[i] = new book;

```

```

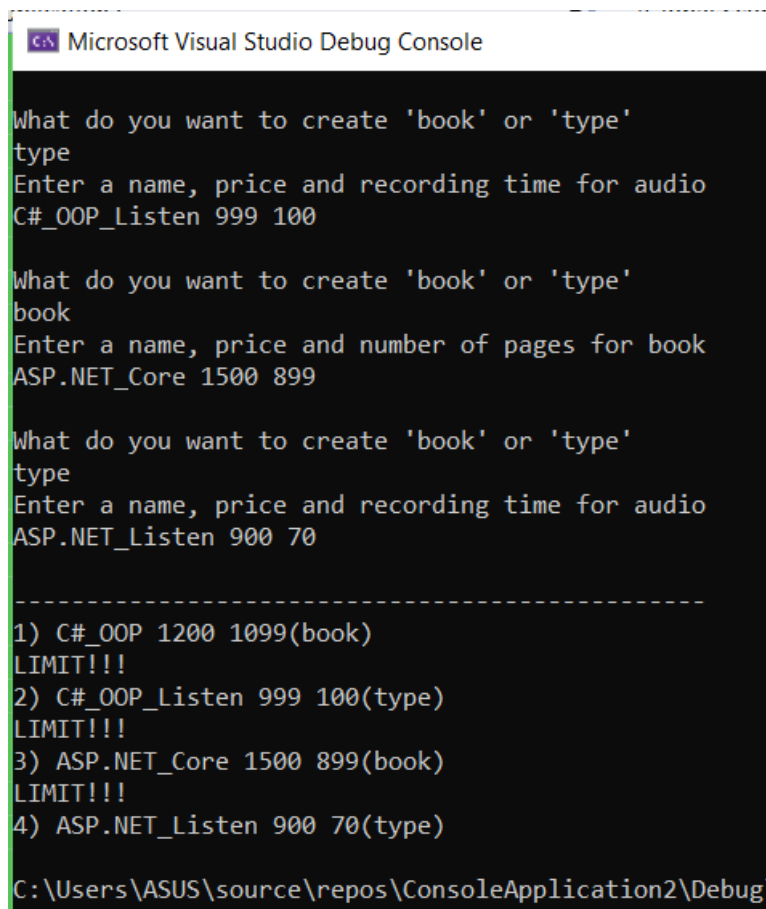
    }
    else if (choice == "type")
    {
        pubarr[i] = new type;
    }

    pubarr[i]->getData();
    cout << endl;
    i++;
}

cout << "-----" << endl;
for (int i = 0; i < size; i++)
{
    cout << i + 1 << ") ";
    pubarr[i]->putdata();
    if (pubarr[i]->isOveersize() == true) {
        cout << "LIMIT!!!" << endl;
    }
}
return 0;
}

```

## РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ



The screenshot shows the Microsoft Visual Studio Debug Console with the following output:

```

What do you want to create 'book' or 'type'
type
Enter a name, price and recording time for audio
C#_OOP_Listen 999 100

What do you want to create 'book' or 'type'
book
Enter a name, price and number of pages for book
ASP.NET_Core 1500 899

What do you want to create 'book' or 'type'
type
Enter a name, price and recording time for audio
ASP.NET_Listen 900 70

-----
1) C#_OOP 1200 1099(book)
LIMIT!!!
2) C#_OOP_Listen 999 100(type)
LIMIT!!!
3) ASP.NET_Core 1500 899(book)
LIMIT!!!
4) ASP.NET_Listen 900 70(type)

C:\Users\ASUS\source\repos\ConsoleApplication2\Debug

```

# Завдання 3

2. Створити клас ТРІЙКА ЧИСЕЛ. Визначити віртуальну функцію піднесення кожного числа до квадрату. Створити похідні класи ТРИКУТНИК (з полями-сторонами), ПАРАЛЕЛОГРАМ (з полями: дві сторони, площа) зі своїми функціями обчислення кутів. Для перевірки використати масив вказівників на об'єкти базового класу, яким присвоїти адреси об'єктів похідних класів.

## КОД ПРОГРАМИ

```
#include <iostream>
#include <math.h>
using namespace std;

class pairThreeNumbers
{
private:
    int number1, number2, number3;
public:
    pairThreeNumbers(int number1, int number2, int number3)
    {
        this->number1 = number1;
        this->number2 = number2;
        this->number3 = number3;
    }
    pairThreeNumbers() {}

    virtual int number1ToSquare() { return number1 * number1; }
    virtual int number2ToSquare() { return number2 * number2; }
    virtual int number3ToSquare() { return number3 * number3; }
    virtual void showAnglesParallelogram() = 0;
};

class Triangle : public pairThreeNumbers
{
private:
    int side1, side2, side3;
public:
    Triangle(int side1, int side2, int side3)
    {
        this->side1 = side1;
        this->side2 = side2;
        this->side3 = side3;
    }
};
```



```

class Parallelogram : public pairThreeNumbers
{
private:
    int side1, side2;
    double square;

public:
    Parallelogram(int side1, int side2, double square)
    {
        this->side1 = side1; this->side2 = side2;
        this->square = square;
    }
private:
    double calculateSinAngle1()
    {
        double sinAngle1 = (square) / (side1 * side2);
        double angle1 = (asin(sinAngle1) * 180.0 / 3.1415);
        return angle1;
    }
    double calculateSinAngle2()
    {
        double angle2 = (360 - (calculateSinAngle1() * 2)) / 2;
        return angle2;
    }
public:
    void showAnglesParallelogram()
    {
        cout << "Angle1 = " << calculateSinAngle1() << " ";
        cout << "Angle2 = " << calculateSinAngle2() << endl;
    }
};

int main()
{
    cout << "Enter side1 side2 and the square of the parallelogram three times" << endl
    << endl;

    int side1, side2; double square;
    cin >> side1 >> side2 >> square;

    Parallelogram myParalelogram1(side1, side1, square);

    cin >> side1 >> side2 >> square;

    Parallelogram myParalelogram2(side1, side1, square);

    cin >> side1 >> side2 >> square;
    cout << "-----" << endl;

    Parallelogram myParalelogram3(side1, side1, square);

    const int size = 3;
    pairThreeNumbers* arrParalelogram[size];
    arrParalelogram[0] = &myParalelogram1;
    arrParalelogram[1] = &myParalelogram2;
    arrParalelogram[2] = &myParalelogram3;
}

```

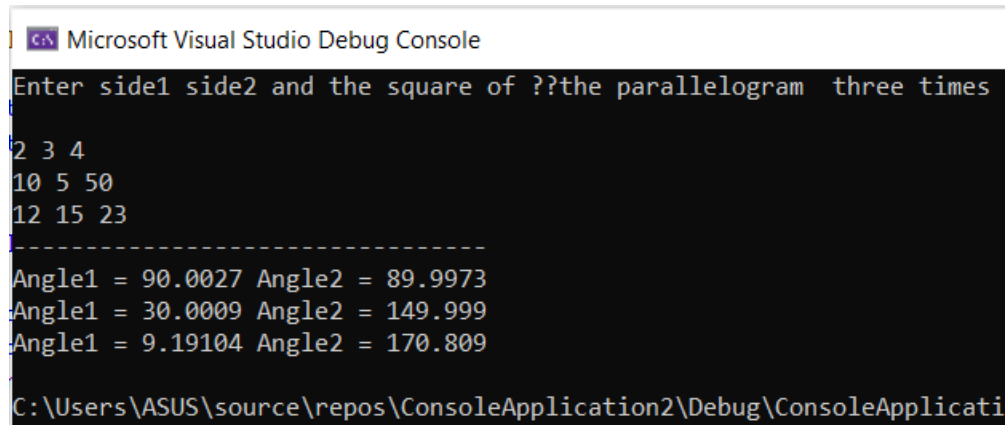
```

    for (int i = 0; i < size; i++)
    {
        arrParalelogram[i]->showAnglesParallelogram();
    }

    return 0;
}

```

## РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ



```

Microsoft Visual Studio Debug Console
Enter side1 side2 and the square of ??the parallelogram three times
2 3 4
10 5 50
12 15 23
-----
Angle1 = 90.0027 Angle2 = 89.9973
Angle1 = 30.0009 Angle2 = 149.999
Angle1 = 9.19104 Angle2 = 170.809
C:\Users\ASUS\source\repos\ConsoleApplication2\Debug\ConsoleApplicati

```

**Висновок:** Практично ознайомився з поняттям поліморфізму, його застосуванням та вивчив механізм його реалізації за допомогою віртуальних функцій