### Кам’янець-Подільський національний університет імені Івана Огієнка

### КАФЕДРА КОМП’ЮТЕРНИХ НАУК

### Навчальна дисципліна «Об’єктно-орієнтоване програмування»

### ЛАБОРАТОРНА РОБОТА #0104

## Тема:

## Спадкування

Варіант №2

**Виконав**:  
студент 1-го курсу  
групи KNms1-B23  
Веретко С. І.

**Прийняла**:  
старший викладач,  
Мястковська М.О.

Кам’янець-Подільський – 2023

1. Теоретичні відомості

Поліморфізм – наступний з трьох принципів ООП. Його застосування дозволяє створювати у похідних класах методи, які мають той самий інтерфейс, що й методи базового класу, проте відмінну функціональність.

Можлива ситуація, коли у базовому класі неможливо реалізувати деякий метод. Проте зрозуміло, що подібний метод зі своєю особливою функціональністю буде присутній у похідних класах. В такому випадку є сенс визначити «порожній» віртуальний метод у базовому класі (свого роду «заглушку»), визначивши його інтерфейс, а у похідних класах слушним чином визначити коди відповідних методів, що його заміщають. Засіб, який примушує всі похідні класи обов’язково замістити такий «невизначений» метод, полягає у використанні модифікатору **abstract**:

abstract <тип\_результату> <ідентифікатор\_методу> (<параметри\_методу>);

Такий метод автоматично є віртуальним і його ***не треба*** додатково помічати службовим словом **virtual**.

Абстрактний метод, таким чином, взагалі не має тіла. Абстрактними можуть бути не лише методи, але й властивості. Клас, який містить принаймні один абстрактний член класу, теж має бути визначений як **abstract**, адже від нього не можна утворювати екземпляри – такі класи використовуються лише для спадкування. Класичним прикладом демонстрації абстрактного класу є клас геометричних форм – не існує загальної формули для визначення площі абстрактної геометричної фігури. Проте всім відомо як визначити площі, наприклад, квадрата або прямокутника чи кола.

# 2. Завдання для виконання

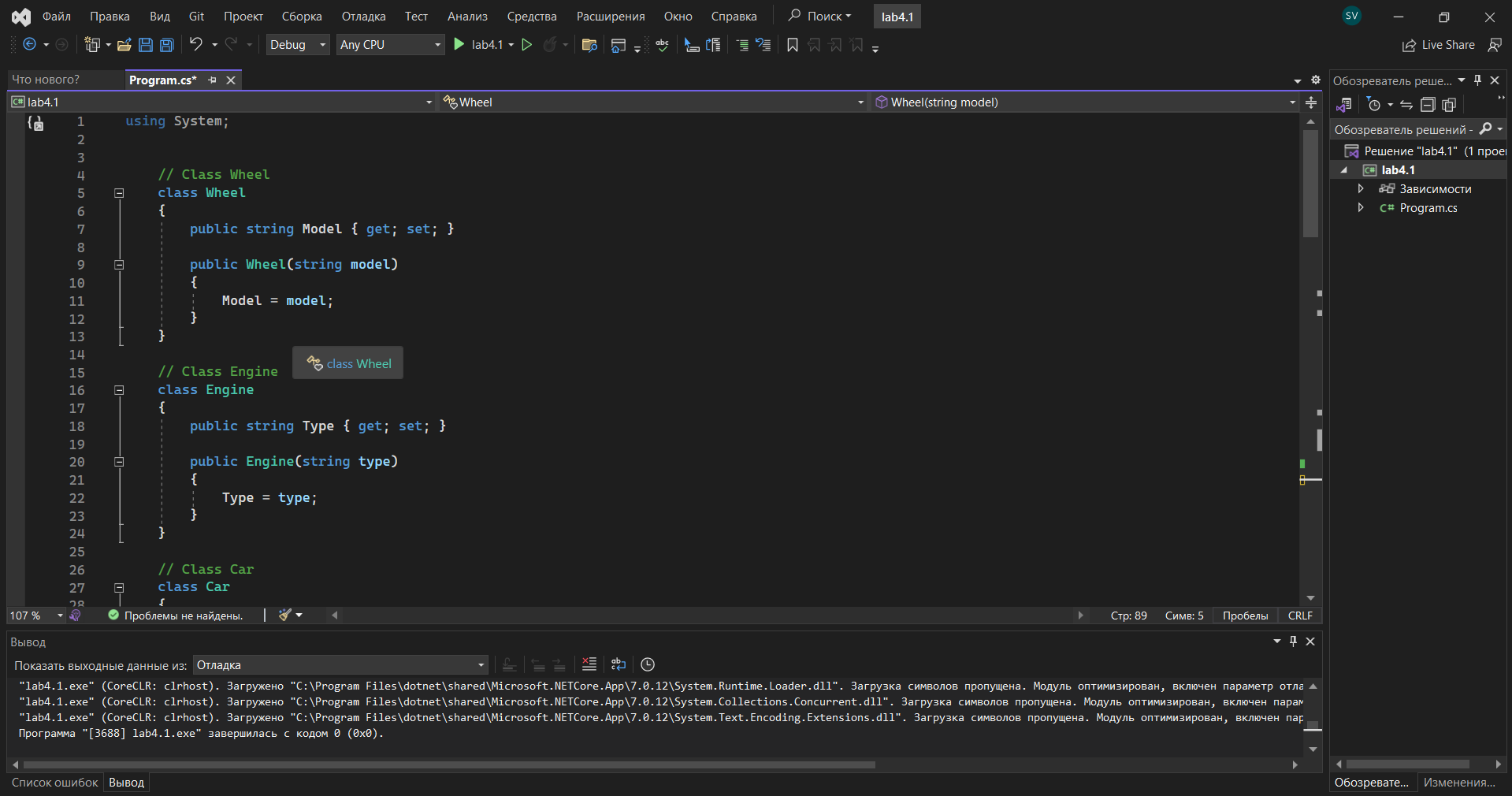
Завдання 1

Створити об'єкт класу Автомобіль, використовуючи класи Колесо, Двигун. Методи: їхати, заправлятися, міняти колесо, вивести на консоль марку автомобіля.

Завдання 2

Створити клас TNumber з віртуальними методами для знаходження суми цифр та знаходження першої/останньої цифри. На основі цього класу створити класи- нащадки TIntNumber та TRealNumber, у яких реалізовано перевизначені віртуальні методи. Створити *m* об’єктів цілих чисел та *n* об’єктів дійсних чисел (дані згенерувати випадковим чином). Знайти суму перших цифр цілих чисел та суму останніх цифр дійсних чисел.

3. Послідовність виконання завданнь



4. Лістинг програми

Завдання 4.1  
using System;

// Class Wheel

class Wheel

{

public string Model { get; set; }

public Wheel(string model)

{

Model = model;

}

}

// Class Engine

class Engine

{

public string Type { get; set; }

public Engine(string type)

{

Type = type;

}

}

// Class Car

class Car

{

public string Brand { get; set; }

public Wheel FrontWheel { get; set; }

public Wheel RearWheel { get; set; }

public Engine Engine { get; set; }

public Car(string brand, Wheel frontWheel, Wheel rearWheel, Engine engine)

{

Brand = brand;

FrontWheel = frontWheel;

RearWheel = rearWheel;

Engine = engine;

}

public void Drive()

{

Console.WriteLine("The car is driving.");

}

public void Refuel()

{

Console.WriteLine("The car is refueling.");

}

public void ChangeWheel(Wheel oldWheel, Wheel newWheel)

{

if (oldWheel == FrontWheel)

{

FrontWheel = newWheel;

}

else if (oldWheel == RearWheel)

{

RearWheel = newWheel;

}

Console.WriteLine("The wheel has been replaced.");

}

public void PrintBrand()

{

Console.WriteLine($"Car brand: {Brand}");

}

}

public class Program

{

static void Main()

{

// Example usage

Wheel frontWheel = new Wheel("Summer");

Wheel rearWheel = new Wheel("Summer");

Wheel newWheel = new Wheel("Summer");

Engine engine = new Engine("Petrol");

Car car = new Car("Mercedes", frontWheel, rearWheel, engine);

car.Drive();

car.Refuel();

car.ChangeWheel(rearWheel,newWheel);

car.PrintBrand();

}

}

Завдання 4.2

using System;

class TNumber

{

protected string number;

public TNumber(string number)

{

this.number = number;

}

public virtual int SumOfDigits()

{

int sum = 0;

foreach (char digit in number)

{

if (char.IsDigit(digit))

{

sum += int.Parse(digit.ToString());

}

}

return sum;

}

public virtual char FirstDigit()

{

foreach (char digit in number)

{

if (char.IsDigit(digit))

{

return digit;

}

}

return '0';

}

public virtual char LastDigit()

{

char last = '0';

foreach (char digit in number)

{

if (char.IsDigit(digit))

{

last = digit;

}

}

return last;

}

}

class TIntNumber : TNumber

{

public TIntNumber(string number) : base(number)

{

}

public override int SumOfDigits()

{

return base.SumOfDigits();

}

}

class TRealNumber : TNumber

{

public TRealNumber(string number) : base(number)

{

}

public override char LastDigit()

{

char last = '0';

foreach (char digit in number)

{

if (char.IsDigit(digit))

{

last = digit;

}

}

return last;

}

}

class Program

{

static void Main(string[] args)

{

Random random = new Random();

int m = 5;

int n = 5;

int sumOfFirstDigits = 0;

int sumOfLastDigits = 0;

TIntNumber[] intNumbers = new TIntNumber[m];

TRealNumber[] realNumbers = new TRealNumber[n];

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

{

int randomInt = random.Next(1, 1000);

intNumbers[i] = new TIntNumber(randomInt.ToString());

sumOfFirstDigits += intNumbers[i].SumOfDigits();

}

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

{

double randomReal = random.NextDouble() \* 1000;

realNumbers[i] = new TRealNumber(randomReal.ToString("0.00"));

sumOfLastDigits += int.Parse(realNumbers[i].LastDigit().ToString());

}

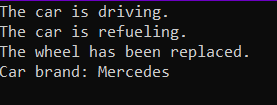
Console.WriteLine("Сума перших цифр цілих чисел: " + sumOfFirstDigits);

Console.WriteLine("Сума останніх цифр дійсних чисел: " + sumOfLastDigits);

}

}

5. Результат роботи програми


6. [xanax4rl/lab (github.com)](https://github.com/xanax4rl/lab)