Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний  інститут імені Ігоря Сікорського"

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 1 з дисципліни

«Основи програмування. Частина 2. Модульне програмування»

«ДОСЛІДЖЕННЯ МЕХАНІЗМУ УСПАДКУВАННЯ»

Варіант 23

Виконав студент Ткаченко Костянтин Олександрович

 (шифр, прізвище, ім'я, по батькові)

Перевірив викладач \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

 ( прізвище, ім'я, по батькові)

Київ 2024

**ДОСЛІДЖЕННЯ МЕХАНІЗМУ УСПАДКУВАННЯ**

**Мета роботи** – дослідити механізм успадкування класів.

**Варіант 23**

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Опис базового класу** | **Опис похідного класу** | **Завдання** |
| **23** | Фігури:  координати вершин,  конструктор з  параметрами, метод  обчислення довжини  сторони | Трапеція:  конструктор з  параметрами, методи  обчислення площі та  периметра, методи  отримання даних  об’єкту | Фігури:  координати вершин,  конструктор з  параметрами, метод  обчислення довжини  сторони |

**Код программи:**

namespace Laba4;

class Program

{

static void Main()

{

var trapezoid = new Trapezoid((1, 1), (4, 1), (5, 5), (2, 5));

Console.WriteLine(trapezoid);

Console.WriteLine("Area: " + trapezoid.Area());

Console.WriteLine("Perimeter: " + trapezoid.Perimeter());

}

}

namespace Laba4;

public class Trapezoid : Figure

{

public Trapezoid(params (double, double)[] vertices) : base(vertices)

{

if (vertices.Length != 4)

throw new ArgumentException("Trapezoid must have exactly four vertices.");

}

public double Area()

{

double area = 0;

for (int i = 0; i < vertices.Length; i++)

{

int next = (i + 1) % vertices.Length;

area += vertices[i].X \* vertices[next].Y - vertices[next].X \* vertices[i].Y;

}

return Math.Abs(area / 2);

}

public double Perimeter()

{

double perimeter = 0;

for (int i = 0; i < vertices.Length; i++)

{

int next = (i + 1) % vertices.Length;

perimeter += SideLength(i, next);

}

return perimeter;

}

public override string ToString()

{

return $"Trapezoid with vertices at ({string.Join(", ", vertices)})";

}

}

namespace Laba4;

public class Figure

{

protected (double X, double Y)[] vertices;

public Figure(params (double, double)[] vertices)

{

this.vertices = vertices;

}

protected double SideLength(int index1, int index2)

{

if (index1 < 0 || index2 < 0 || index1 >= vertices.Length || index2 >= vertices.Length)

{

throw new ArgumentOutOfRangeException("Index out of range");

}

var (x1, y1) = vertices[index1];

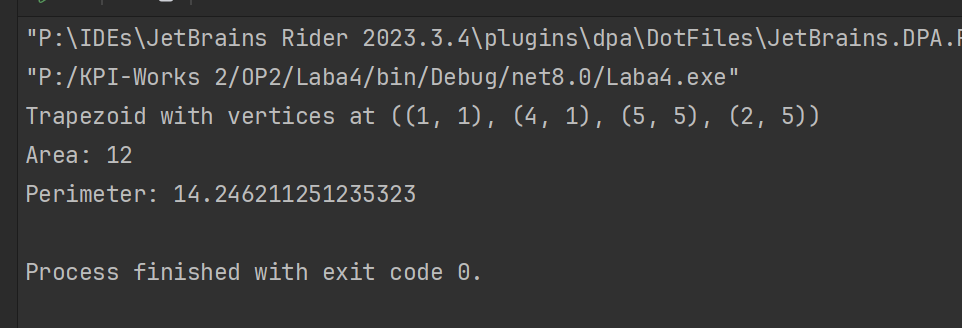
var (x2, y2) = vertices[index2];

return Math.Sqrt(Math.Pow(x2 - x1, 2) + Math.Pow(y2 - y1, 2));

}

}

**Результат роботи:**



**Висновок**

У даному проекті ми створили клас **Trapezoid**, який представляє собою трапецію у двовимірному просторі. Клас має методи для обчислення площі та периметра трапеції на основі заданих вершин.

У методі **Main** ми створили екземпляр трапеції з заданими вершинами, викликали методи для обчислення площі та периметра та вивели результати на консоль.

Цей проект ілюструє використання класів та методів для роботи з геометричними фігурами у мові програмування C#. Крім того, він демонструє використання наслідування для створення ієрархії класів, де клас **Trapezoid** успадковує методи з класу **Figure**, що містить загальні функції для роботи з геометричними фігурами.