

Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»



ЗВІТ
про виконання лабораторної роботи № 3

Виконав:
студент гр. 122-19-3
Файнштейн Д. В.

Перевірили:
доц. Приходченко С.Д.
ас. Шевцова О.С.

Дніпро
2020

Лабораторна робота №3-OOP

Вивчення успадкування як методики повторного використання коду в класах C#

Варіант 24 (У роботі – 4)

Цілі роботи

- навчитися будувати найпростіші класові моделі;
- оволодіти методикою автоматичної генерації класів C# з діаграм класів;
- створити визначення функцій-членів класів з використанням додаткових засобів відображення графіки в консолі.

Умова задачі

Створити класову модель, яка включає в себе абстрактний клас CGraphicsObject, його спадкоємець - базовий клас CPoint, і клас-спадкоємець від CPoint, що представляє собою графічний об'єкт відповідно до заданого варіантом. Всі класи, за винятком абстрактного, повинні містити конструктори, деструктори, відповідні set- і get-методи, а також метод Show, який може мати різну сигнатуру для різних класів і повинен виводити на екран в текстовому вигляді інформацію про об'єкт і його предків. Абстрактний клас повинен містити декларацію полів і методів.

В роботі обов'язково застосування засобів автоматизації; в підсумковому звіті необхідно показати скріншот з діаграмою класів.

Примітка: абстрактний клас не потребує реалізації методів

4

Эллиптический сектор

Лістинг програми

CGraphicsObject.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Lab3_Fainshtein
{
    public abstract class CGraphicsObject
    {
        protected double radius1;
        protected double radius2;
        protected double angleDeg;
    }
}
```

```

        protected double width;
        protected double length;
        protected double x;
        protected double y;
        protected double arcLength;

        public abstract string Show();
    }
}

```

CPoint.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Lab3_Fainshtein
{
    public class CPoint : CGraphicsObject
    {
        public CPoint()
        {
            x = 0;
            y = 0;
        }

        ~CPoint()
        {
            Console.WriteLine("Координаты удалены");
        }

        public double X
        {
            get { return x; }
            set { x = value; }
        }

        public double Y
        {
            get { return y; }
            set { y = value; }
        }

        public override string Show()
        {
            return "Координаты точки: X = " + X + "Y = " + Y;
        }
    }
}

```

EllipticalSector.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Lab3_Fainshtein
{
    public class EllipticalSector : CPoint
    {
        public EllipticalSector(double rad1, double rad2, double arcS, double angleD)
        {
            radius1 = rad1;

```

```

        radius2 = rad2;
        arcLength = arcS;
        angleDeg = angleD;
    }

    ~EllipticalSector()
    {
        Console.WriteLine("Эллиптический сектор удален");
    }

    public double Radius1 // Первый радиус
    {
        get { return radius1; }
        set { radius1 = value; }
    }

    public double Radius2 // Второй радиус
    {
        get { return radius2; }
        set { radius2 = value; }
    }

    public double AngleDeg // Градус угла
    {
        get { return angleDeg; }
        set { angleDeg = value; }
    }

    public double ArcLength // Длина дуги
    {
        get { return arcLength; }
        set { arcLength = value; }
    }

    public override string Show()
    {
        return "Первый радиус сектора = " + Radius1 + ", второй радиус сектора = " +
Radius2 + ", дуга = " + ArcLength
            + ", угол = " + AngleDeg + ", координаты: " + X + ", " + Y;
    }
}
}

```

Program.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Lab3_Fainshtein
{
    class Program
    {
        static void Main(string[] args)
        {
            string t;
            EllipticalSector ellipticalSector = new EllipticalSector(12, 15, 124, 45);
            ellipticalSector.X = 12;
            ellipticalSector.Y = 34;

            t = ellipticalSector.Show();
            Console.WriteLine(t);
        }
    }
}

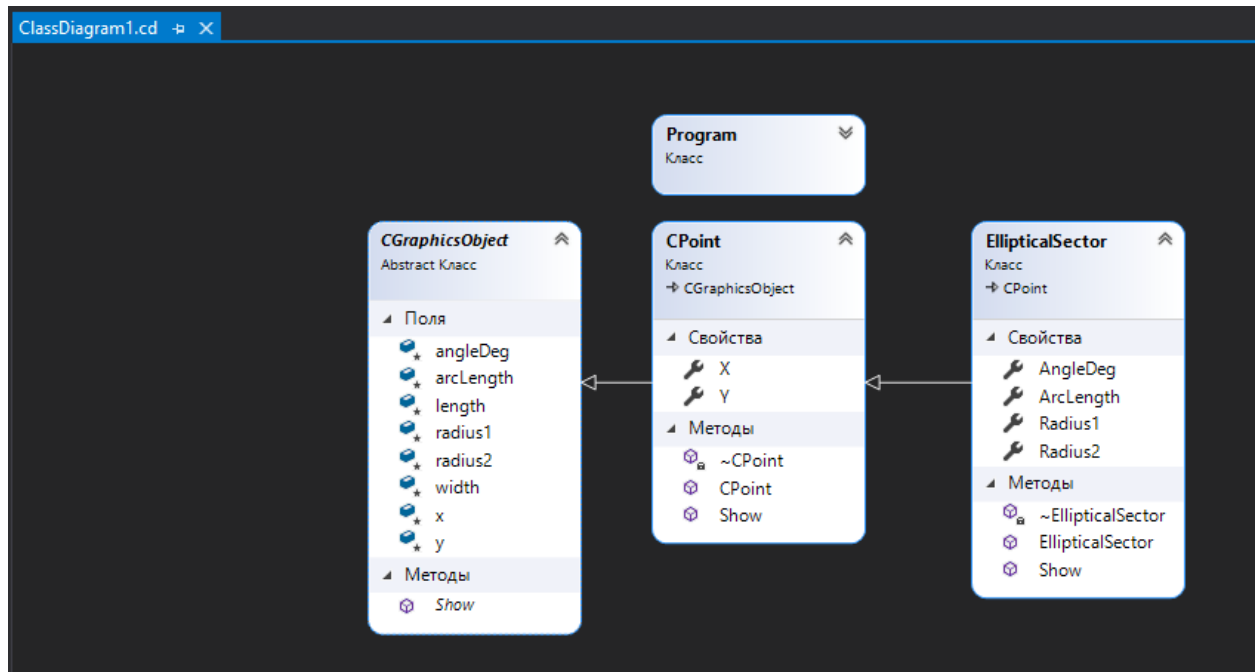
```

```

    Console.ReadLine();
}
}
}

```

Результат



Первый радиус сектора = 12, второй радиус сектора = 15, дуга = 124, угол = 45, координаты: 12, 34

Висновок: навчився будувати найпростіші класові моделі, оволодів методикою автоматичної генерації класів C# з діаграм класів, створив визначення функцій-членів класів з використанням додаткових засобів відображення графіки в консолі.