

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



Практична робота №3 (2 вар.)
з дисципліни “Об’єктно-орієнтоване програмування”

Виконав:

Студент групи

122-19-3

Бородін П.Є.

м.Дніпро

2020

Тема роботи: Изучение наследования как методики повторного использования кода в классах C#

Мета:

- ☐ научиться строить простейшие классовые модели;
- ☐ овладеть методикой автоматической генерации классов C# из диаграмм классов;
- ☐ создать определение функций-членов классов с использованием дополнительных средств отображения графики в консоли.

Технічне завдання: Создать классовую модель, которая включает в себя абстрактный класс CGraphicsObject, его наследник - базовый класс CPoint, и класс-наследник от CPoint, представляющий собой графический объект в соответствии с заданным вариантом. Все классы, за исключением абстрактного, должны содержать конструкторы, деструкторы, соответствующие set- и get-методы, а также метод Show, который может иметь различную сигнатуру для разных классов и должен выводить на экран в текстовом виде информацию об объекте и его предках. Абстрактный класс должен содержать декларацию полей и методов.

В работе обязательно применение средств автоматизации; в итоговом отчете необходимо показать скриншот с диаграммой классов.

2 варіант - Дуга окружности

Лістинг:

CGraphicsObject.cs

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

namespace Lab_3
{
    public abstract class CGraphicsObject
    {
        double radius;
        double angle;

        public abstract void SetRadius(double radius);
        public abstract void SetAngle(double angle);
        public abstract double GetRadius();
        public abstract double GetAngle();
    }
}
```

CPoint.cs

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

namespace Lab_3
{
    public class CPoint : CGraphicsObject
    {
        public double radius, angle;
        public CPoint(double r, double a)
        {
            radius = r;
            angle = a;
        }

        ~CPoint()
        {
            Console.WriteLine("Об'єкт зруйновано");
        }

        public override void SetRadius(double r)
        {
            radius = r;
        }
        public override double GetRadius()
        {
            return radius;
        }
        public override void SetAngle(double a)
        {
            angle = a;
        }
        public override double GetAngle()
        {
            return angle;
        }

        public void Show()
        {
            Console.WriteLine("Радіус дуги = {0,5:f3}, Центральний кут (у радіанах) = {0,5:f3}", radius, angle);
        }
    }
}
```

CircleArc.ps

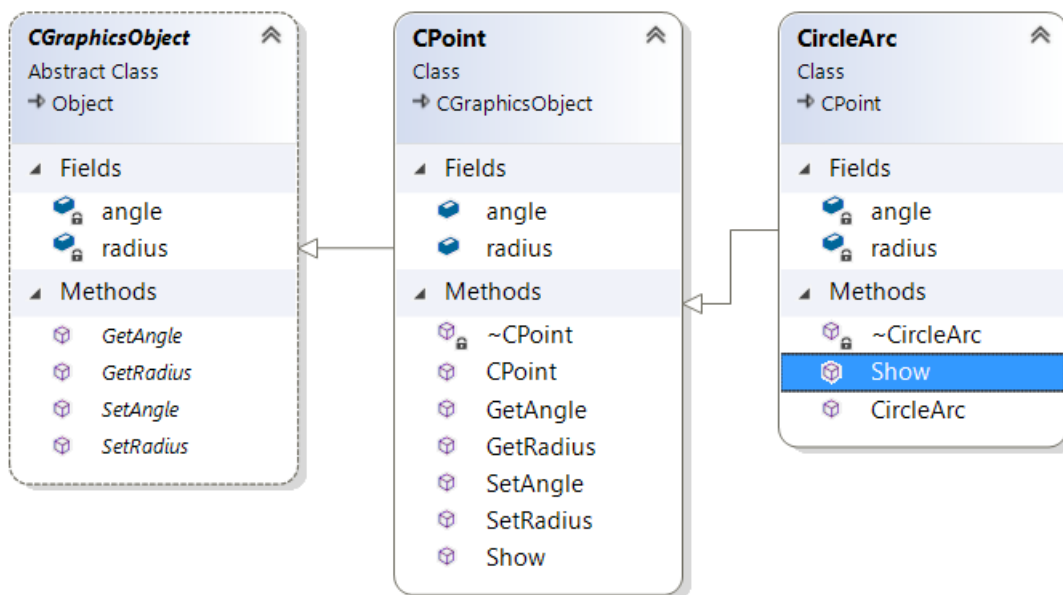
```
using System;
using System.Collections.Generic;
using System.Text;

namespace Lab_3
{
    public class CircleArc : CPoint
    {
        double radius, angle;
        public CircleArc(double r, double a) : base(r, a) { }

        ~CircleArc()
        {
            Console.WriteLine("Об'єкт зруйновано");
        }

        public void Show()
        {
            Console.WriteLine("Радіус дуги = {0,5:f3}, Центральний кут (у радіанах) = {0,5:f3}, Довжина дуги = {0,5:f3}", radius, angle, radius * angle);
        }
    }
}
```

Скріншоти:



Висновок:

- ☐ навчився будувати простіші класові моделі;
- ☐ володів методикою автоматичної генерації класів C# з діаграм класів;

□ создал определение функций-членов классов с использованием дополнительных средств отображения графики в консоли.