



**Московский государственный технический университет**

**им. Н.Э. Баумана**

**(МГТУ им. Н.Э. Баумана)**

**Кафедра «Системы обработки информации и управления» (ИУ5)**

Лабораторная работа № 3

По дисциплине: «Парадигмы и конструкции языков программирования»

«КЛАССЫ ГЕОМЕТРИЧЕСКИХ ФИГУР»

Выполнила: Вешторт Е. С.,

студентка группы ИУ5-32Б

Проверил: Нардид А. Н.

г. Москва 2023 г.

Задание: создать родительский класс “геометрическая фигура” и на основании него создать классы прямоугольник, квадрат, круг. Реализовать простую коллекцию и простой стек на основе односвязного списка для этих геометрических фигур. Реализовать сортировку стека и коллекции.

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

// Абстрактный класс "Геометрическая фигура"
public abstract class GeometricFigure : IComparable<GeometricFigure>, IPrint
{
    public abstract double CalculateArea();

    public int CompareTo(GeometricFigure other)
    {
        return this.CalculateArea().CompareTo(other.CalculateArea());
    }

    public abstract string FigureType { get; }

    public override string ToString()
    {
        return $"{FigureType} с площадью {CalculateArea()}";
    }

    public void Print()
    {
        Console.WriteLine(ToString());
    }
}

// Класс "Прямоугольник"
public class Rectangle : GeometricFigure
{
    private double width;
    private double height;

    public Rectangle(double width, double height)
    {
        this.width = width;
        this.height = height;
    }

    public override double CalculateArea()
    {
        return width * height;
    }

    public override string FigureType => "Прямоугольник";
}

// Класс "Квадрат"
public class Square : Rectangle
{
    public Square(double side) : base(side, side) { }

    public override string FigureType => "Квадрат";
}

// Класс "Круг"
public class Circle : GeometricFigure
{

```

```

    private double radius;

    public Circle(double radius)
    {
        this.radius = radius;
    }

    public override double CalculateArea()
    {
        return Math.PI * radius * radius;
    }

    public override string FigureType => "Круг";
}

// Интерфейс для печати
public interface IPrint
{
    void Print();
}

// Класс "Простая коллекция"
public class SimpleCollection<T> : IEnumerable<T> where T : IComparable<T>
{
    protected List<T> items;

    public SimpleCollection()
    {
        items = new List<T>();
    }

    public void Add(T item)
    {
        items.Add(item);
    }

    public void Sort()
    {
        items.Sort();
    }

    public IEnumerator<T> GetEnumerator()
    {
        return items.GetEnumerator();
    }

    IEnumerator IEnumerable.GetEnumerator()
    {
        return GetEnumerator();
    }
}

// Односвязный узел
public class Node<T>
{
    public T Data { get; set; }
    public Node<T> Next { get; set; }

    public Node(T data)
    {
        Data = data;
        Next = null;
    }
}

```

```

// Класс "Простой стек" на основе односвязного списка
public class SimpleStack<T> : SimpleCollection<T> where T : IComparable<T>
{
    private Node<T> top;

    public void Push(T element)
    {
        Node<T> newNode = new Node<T>(element);
        newNode.Next = top;
        top = newNode;
    }

    public T Pop()
    {
        if (top == null)
        {
            throw new InvalidOperationException("Стек пуст.");
        }

        T data = top.Data;
        top = top.Next;

        return data;
    }
}

class Program
{
    static void Main()
    {
        // Создаем объекты
        GeometricFigure rectangle = new Rectangle(5, 10);
        GeometricFigure square = new Square(7);
        GeometricFigure circle = new Circle(3);

        // Инициализируем коллекции
        SimpleCollection<GeometricFigure> figureList = new
SimpleCollection<GeometricFigure>();
        SimpleStack<GeometricFigure> figureStack = new
SimpleStack<GeometricFigure>();

        // Добавляем объекты в список
        figureList.Add(rectangle);
        figureList.Add(square);
        figureList.Add(circle);

        // Добавляем объекты в стек
        figureStack.Push(rectangle);
        figureStack.Push(square);
        figureStack.Push(circle);

        // Выводим список
        Console.WriteLine("Список:");
        foreach (var figure in figureList)
        {
            figure.Print();
        }

        // Сортируем коллекцию
        figureList.Sort();

        // Сортируем стек
        figureStack.Sort();
    }
}

```

```

// Выводим результаты
Console.WriteLine("\nСортированный список:");
foreach (var figure in figureList)
{
    figure.Print();
}

Console.WriteLine("\nСортированный стек:");
while (true)
{
    try
    {
        GeometricFigure figure = figureStack.Pop();
        figure.Print();
    }
    catch (InvalidOperationException ex)
    {
        Console.WriteLine(ex.Message);
        break;
    }
}
}
}

```

Список:

Прямоугольник с площадью 50

Квадрат с площадью 49

Круг с площадью 28.274333882308138

Сортированный список:

Круг с площадью 28.274333882308138

Квадрат с площадью 49

Прямоугольник с площадью 50

Сортированный стек:

Круг с площадью 28.274333882308138

Квадрат с площадью 49

Прямоугольник с площадью 50

Стек пуст.