

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 № 2
з дисципліни «Основи програмування – 2.
Методології програмування»

«Класи та об'єкти»

Варіант 13

Виконав студент ПІ-13 Жмайло Дмитро Олександрович
(шифр, прізвище, ім'я, по батькові)

Перевірив Вечерковська Анастасія Сергіївна
(прізвище, ім'я, по батькові)

Київ 2022

Лабораторна робота 2

Класи та об'єкти

Варіант 13

Розробити клас "сфера в просторі", яка задається радіусом та координатами центру. Створити масив об'єктів даного класу. Визначити номери тих куль, в які попадає введена користувачем точка.

Код програми

C#

Program.cs

```
using System;

using System.Collections.Generic;

namespace Lab2CSharp
{
    class Program
    {
        static void Main(string[] args)
        {
            int number = Operations.InputInteger("Enter number of spheres: ");
            Console.WriteLine();
            Sphere[] arrayOfSpheres = Operations.InputSpheres(number);
            Console.WriteLine("Enter coordinates of the point: ");
            double xDotCoordinate = Operations.InputCoordinate("x");
            double yDotCoordinate = Operations.InputCoordinate("y");
            double zDotCoordinate = Operations.InputCoordinate("z");
            Console.WriteLine();

            List<int> numbers = Operations.GetNumberOfSpheres(arrayOfSpheres,
xDotCoordinate, yDotCoordinate, zDotCoordinate, out bool[] result);

            Operations.ShowResult(arrayOfSpheres, xDotCoordinate, yDotCoordinate,
zDotCoordinate, result, numbers);

            Console.WriteLine();
            Operations.ShowNumbers(numbers);
            Console.ReadKey();
        }
    }
}
```

Operations.cs

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

namespace Lab2CSharp
{
    class Operations
    {
        public static Sphere[] InputSpheres(int number)
        {
            Sphere[] result = new Sphere[number];
            for (int i = 0; i < number; i++)
            {
                Console.WriteLine($"Enter info about {i}) sphere: ");
                double x = InputCoordinate("x");
                double y = InputCoordinate("y");
                double z = InputCoordinate("z");
                double radius = InputInteger("\tEnter radius: ");
                result[i] = new Sphere(x, y, z, radius);
                Console.WriteLine();
            }
            return result;
        }

        public static int InputInteger(string message)
        {
            Console.Write(message);
            while (true)
            {
                string input = Console.ReadLine();
                if (int.TryParse(input, out int result) && result > 0)
                {
                    return result;
                }
            }
            else
```

```

        {
            Console.WriteLine("\tWrong input. Try again: ");
        }
    }
}

```

```

public static double InputCoordinate(string message)
{
    Console.WriteLine($"{\tEnter {message} coordinate: ");
    while (true)
    {
        string input = Console.ReadLine();
        if (double.TryParse(input, out double result))
        {
            return result;
        }
        else
        {
            Console.WriteLine("\tWrong input. Try again: ");
        }
    }
}

```

```

public static List<int> GetNumberOfSpheres(Sphere[] spheres, double x, double y,
double z, out bool[] result)
{
    List<int> numbers = new List<int>();
    result = new bool[spheres.Length];
    for (int i = 0; i < spheres.Length; i++)
    {
        if (spheres[i].isDotInTheSphere(x, y, z))
        {
            result[i] = true;
            numbers.Add(i);
        }
        else
        {

```

```

        result[i] = false;
    }
}
return numbers;
}

public static void ShowResult(Sphere[] spheres, double xDot, double yDot, double
zDot, bool[] result, List<int>numbers)
{
    Console.WriteLine("Result: ");
    Console.Write($"Coordinates of the point are: ");
    Console.ForegroundColor = ConsoleColor.DarkGreen;
    Console.WriteLine($"x =\t{xDot}, y =\t{yDot}, z =\t{zDot}");
    Console.ResetColor();

    for (int i = 0; i < spheres.Length; i++)
    {
        Console.Write($"Coordinates of the {i}) sphere are: ");
        Console.ForegroundColor = ConsoleColor.Blue;
        Console.Write($"x =\t{spheres[i].GetXCoordinate()}, y
=\t{spheres[i].GetYCoordinate()}, z =\t{spheres[i].GetZCoordinate()}, r
=\t{spheres[i].GetRadius()}");
        Console.ResetColor();

        if (spheres[i].isDotInTheSphere(xDot, yDot, zDot))
        {
            Console.ForegroundColor = ConsoleColor.Green;
            Console.WriteLine("\tthe point is in the sphere");
            Console.ResetColor();
        }
        else
        {
            Console.ForegroundColor = ConsoleColor.Red;
            Console.WriteLine("\tthe point is outside of the sphere");
            Console.ResetColor();
        }
    }
}
}

```

```
public static void ShowNumbers(List<int>numbers)
{
    Console.Write("Sequence numbers of suitable spheres are: ");
    if (numbers.Count == 0)
    {
        Console.ForegroundColor = ConsoleColor.Red;
        Console.Write("None");
    }
    Console.ForegroundColor = ConsoleColor.Green;
    for (int k = 0; k < numbers.Count; k++)
    {
        Console.Write(numbers[k] + ") ");
    }
    Console.ResetColor();
}
}
```

Sphere.cs

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

namespace Lab2CSharp
{
    class Sphere
    {
        private double xCenterCoordinate, yCenterCoordinate, zCenterCoordinate, radius;

        public double GetXCoordinate()
        {
            return xCenterCoordinate;
        }

        public double GetYCoordinate()
        {
            return yCenterCoordinate;
        }

        public double GetZCoordinate()
        {
            return zCenterCoordinate;
        }

        public double GetRadius()
        {
            return radius;
        }

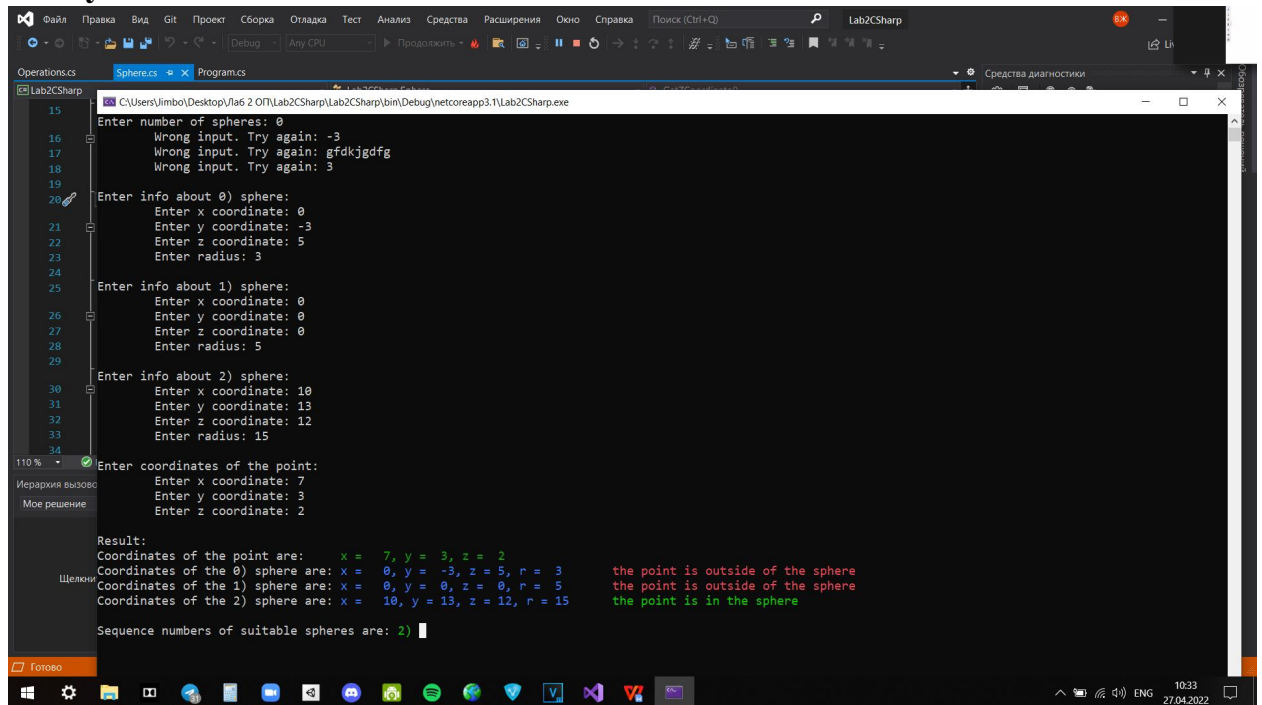
        public Sphere(double xCenterCoordinate, double yCenterCoordinate, double
zCenterCoordinate, double radius)
        {
            this.xCenterCoordinate = xCenterCoordinate;
            this.yCenterCoordinate = yCenterCoordinate;
            this.zCenterCoordinate = zCenterCoordinate;
        }
    }
}
```

```
        this.radius = radius;
    }

    public bool isDotInTheSphere(double x, double y, double z)
    {
        bool result = Math.Pow(x - xCenterCoordinate, 2) + Math.Pow(y -
yCenterCoordinate, 2) + Math.Pow(z - zCenterCoordinate, 2) <= Math.Pow(radius, 2);

        return result;
    }
}
```


Тестування:



```
Enter number of spheres: 0
Wrong input. Try again: -3
Wrong input. Try again: gfdkjgdfg
Wrong input. Try again: 3
Enter info about 0) sphere:
Enter x coordinate: 0
Enter y coordinate: -3
Enter z coordinate: 5
Enter radius: 3
Enter info about 1) sphere:
Enter x coordinate: 0
Enter y coordinate: 0
Enter z coordinate: 0
Enter radius: 5
Enter info about 2) sphere:
Enter x coordinate: 10
Enter y coordinate: 13
Enter z coordinate: 12
Enter radius: 15
Enter coordinates of the point:
Enter x coordinate: 7
Enter y coordinate: 3
Enter z coordinate: 2

Result:
Coordinates of the point are: x = 7, y = 3, z = 2
Coordinates of the 0) sphere are: x = 0, y = -3, z = 5, r = 3 the point is outside of the sphere
Coordinates of the 1) sphere are: x = 0, y = 0, z = 0, r = 5 the point is outside of the sphere
Coordinates of the 2) sphere are: x = 10, y = 13, z = 12, r = 15 the point is in the sphere
Sequence numbers of suitable spheres are: 2)
```

Висновки:

На цій лабораторній роботі я застосував на практиці знання щодо створення та роботою з класами та об'єктами й побачив їх програмну реалізацію на мові програмування C#.