

Министерство образования Республики Беларусь
Учреждение образования
«Брестский Государственный технический университет»
Кафедра ИИТ

Лабораторная работа №3

По дисциплине «Современные платформы программирования»

Выполнил:

Студент 3 курса

Группы ПО-8

Лобарев А.М.

Проверил:

Крощенко А.А.

Брест 2024

Цель работы: научиться создавать и использовать классы.

Вариант 16

Задание 1

Множество вещественных чисел ограниченной мощности – Предусмотреть возможность объединения двух множеств, вывода на печать элементов множества, а также метод, определяющий, принадлежит ли указанное значение множеству. Класс должен содержать методы, позволяющие добавлять и удалять элемент в/из множества. Конструктор должен позволить создавать объекты с начальной инициализацией. Мощность множества задается при создании объекта. Реализацию множества осуществить на базе одномерного массива. Реализовать метод equals, выполняющий сравнение объектов данного типа.

Код:

```
public class RealNumberSet
{
    private double[] elements;
    private int capacity;
    private int count;

    public RealNumberSet(int capacity)
    {
        this.capacity = capacity;
        elements = new double[capacity];
        count = 0;
    }

    public bool Add(double element)
    {
        if (count < capacity && !this.Contains(element))
        {
            elements[count++] = element;
            return true;
        }
        return false;
    }

    public bool Remove(double element)
    {
        int index = Array.IndexOf(elements, element);
        if (index >= 0)
        {
            elements[index] = elements[--count];
            return true;
        }
        return false;
    }

    public bool Contains(double element)
    {
        return Array.IndexOf(elements, element) >= 0;
    }

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

    public RealNumberSet Union(RealNumberSet other)
```

```

    {
        RealNumberSet unionSet = new RealNumberSet(this.capacity + other.capacity);
        foreach (double element in this.elements)
        {
            unionSet.Add(element);
        }
        foreach (double element in other.elements)
        {
            if (!unionSet.Contains(element))
            {
                unionSet.Add(element);
            }
        }
        return unionSet;
    }

    public override bool Equals(object? obj)
    {
        RealNumberSet? other = obj as RealNumberSet;
        if (other == null || this.count != other.count)
            return false;

        for (int i = 0; i < this.count; i++)
        {
            if (this.elements[i] != other.elements[i])
                return false;
        }
        return true;
    }

    public override int GetHashCode()
    {
        return base.GetHashCode();
    }

    public override string ToString()
    {
        return $"RealNumberSet with {count} elements: " + string.Join(", ",
elements.Select(x => $"{x}").ToArray(), 0, count);
    }
}

```

```

internal class Program
{
    static void Main()
    {
        CultureInfo.CurrentCulture = CultureInfo.InvariantCulture;

        var set1 = new RealNumberSet(5);
        set1.Add(1.1);
        set1.Add(2.2);
        set1.Add(3.3);

        var set2 = new RealNumberSet(5);
        set2.Add(4.4);
        set2.Add(5.5);
        set2.Add(3.3);

        var unionSet = set1.Union(set2);
        unionSet.Print(); // Выведет 1.1 2.2 3.3 4.4 5.5
    }
}

```

```

        Console.WriteLine(set1.Contains(2.2)); // Выведет true
        Console.WriteLine(set1.Contains(6.6)); // Выведет false

        set1.Remove(2.2);
        set1.Print(); // Выведет 1.1 3.3
    }
}

```

```

Консоль отладки Microsoft \
RealNumberSet with 5 elements: 1.1, 2.2, 3.3, 4.4, 5.5
True
False
RealNumberSet with 2 elements: 1.1, 3.3

C:\Users\Gygabyte\source\repos\Studies\6 sem\SPP\Lab3\Task1\bin\Debug\net8.0\Task1.exe (процесс 21704) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:

```

Задание 2

Автоматизированная система аренды квартир

Составить программу, которая содержит информацию о квартирах, содержащихся в базе данных бюро обмена квартир. Сведения о каждой квартире (Room) содержат:

- количество комнат;
- общую площадь;
- этаж;
- адрес;
- цену аренды.
- сдается ли квартира.

Программа должна обеспечить:

- Формирование списков свободных занятых квартир;
- Поиск подходящего варианта (при равенстве количества комнат и этажа и различии площадей в пределах 10 кв. м.);
- Удаление квартиры из списка свободных квартир и перемещение в список сдаваемых квартир;
- Вывод полного списка.
- Список квартир, имеющих заданное число комнат;
- Список квартир, имеющих заданное число комнат и расположенных на этаже, который находится в заданном промежутке;
- Список квартир, имеющих площадь, превосходящую заданную.

```

public class Room
{
    public int Rooms { get; set; }
    public double Area { get; set; }
    public int Floor { get; set; }
    public string Address { get; set; }
    public double RentPrice { get; set; }
    public bool IsRented { get; set; }

    public Room(int rooms, double area, int floor, string address, double rentPrice,
bool isRented)
    {
        Rooms = rooms;
        Area = area;
        Floor = floor;
        Address = address;
        RentPrice = rentPrice;
        IsRented = isRented;
    }
}

```

```

public class RentalSystem
{
    private List<Room> roomsList = new List<Room>();

    public void AddRoom(Room room)
    {
        roomsList.Add(room);
    }

    public List<Room> GetFreeRooms()
    {
        return roomsList.Where(room => !room.IsRented).ToList();
    }

    public List<Room> GetRentedRooms()
    {
        return roomsList.Where(room => room.IsRented).ToList();
    }

    public Room? FindRoom(int rooms, int floor, double areaTolerance = 10)
    {
        return roomsList.FirstOrDefault(room => room.Rooms == rooms && room.Floor ==
floor && Math.Abs(room.Area - areaTolerance) <= 10);
    }

    public void RentRoom(Room room)
    {
        room.IsRented = true;
    }

    public void RemoveRoom(Room room)
    {
        roomsList.Remove(room);
    }

    public List<Room> GetRoomsByRoomsCount(int rooms)
    {
        return roomsList.Where(room => room.Rooms == rooms).ToList();
    }
}

```

```

        public List<Room> GetRoomsByFloorRange(int rooms, int minFloor, int maxFloor)
        {
            return roomsList.Where(room => room.Rooms == rooms && room.Floor >= minFloor
&& room.Floor <= maxFloor).ToList();
        }

        public List<Room> GetRoomsByArea(double area)
        {
            return roomsList.Where(room => room.Area > area).ToList();
        }

        public void DisplayRooms()
        {
            foreach (var room in roomsList)
            {
                Console.WriteLine($"Address: {room.Address}, Rooms: {room.Rooms}, Area:
{room.Area}, Floor: {room.Floor}, Rent: {room.RentPrice}, Status: {(room.IsRented ?
"Rented" : "Available")}");
            }
        }
    }

internal class Program
{
    static void Main()
    {
        var rentalSystem = new RentalSystem();
        rentalSystem.AddRoom(new Room(3, 70, 5, "123 Main St", 1500, false));
        rentalSystem.AddRoom(new Room(2, 50, 2, "456 Elm St", 1200, true));
        rentalSystem.AddRoom(new Room(1, 30, 1, "789 Oak St", 900, false));

        rentalSystem.DisplayRooms();
    }
}

```

```

Консоль отладки Microsoft Visual Studio
Address: 123 Main St, Rooms: 3, Area: 70, Floor: 5, Rent: 1500, Status: Available
Address: 456 Elm St, Rooms: 2, Area: 50, Floor: 2, Rent: 1200, Status: Rented
Address: 789 Oak St, Rooms: 1, Area: 30, Floor: 1, Rent: 900, Status: Available

C:\Users\Gygabyte\source\repos\Studies\6 sem\SPP\Lab3\Task2\bin\Debug\net8.0\Task2.exe (процесс 21692) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:

```

Вывод: научился создавать и использовать классы.