

Практична 2.

Завдання 1.

```
using System;
```

```
class Program
```

```
{
```

```
    static void Main()
```

```
    {
```

```
        int lastDigit = 5;
```

```
        int upperBound = 10 + lastDigit;
```

```
        Console.WriteLine("enter 3 numbers:");
```

```
        int a = int.Parse(Console.ReadLine());
```

```
        int b = int.Parse(Console.ReadLine());
```

```
        int c = int.Parse(Console.ReadLine());
```

```
        int[] numbers = { a, b, c };
```

```
        Console.WriteLine($"Number interval from [1, {upperBound}]:");
```

```
        foreach (int number in numbers)
```

```
        {
```

```
            if (number >= 1 && number <= upperBound)
```

```

    {
        Console.WriteLine(number);
    }
}
}
}
}

```

The screenshot shows a C# IDE with a file named 'Main.cs'. The code defines a class 'Program' with a static method 'Main()'. Inside 'Main()', it prompts the user to 'enter 3 numbers:' and reads three integers 'a', 'b', and 'c'. It then calculates an 'upperBound' as 10 plus the last digit of 'a'. It creates an array 'numbers' containing 'a', 'b', and 'c'. It then prints a message 'Number interval from [1, {upperBound}]:' and iterates through the 'numbers' array. For each number, it checks if the number is greater than or equal to 1 and less than or equal to 'upperBound'. If so, it prints the number. The output window shows the execution results: 'enter 3 numbers:', '3', '20', '30', 'Number interval from [1, 15]:', and '3'. The execution is successful.

```

3 class Program
4 {
5     static void Main()
6     {
7         int lastDigit = 5;
8         int upperBound = 10 + lastDigit;
9
10        Console.WriteLine("enter 3 numbers:");
11        int a = int.Parse(Console.ReadLine());
12        int b = int.Parse(Console.ReadLine());
13        int c = int.Parse(Console.ReadLine());
14
15        int[] numbers = { a, b, c };
16
17        Console.WriteLine($"Number interval from [1, {upperBound}]");
18        foreach (int number in numbers)
19        {
20            if (number >= 1 && number <= upperBound)
21            {
22                Console.WriteLine(number);
23            }
24        }
25    }
26 }
27

```

Output:

```

mono /tmp/VXD9AlIIn4.exe
enter 3 numbers:
3
20
30
Number interval from [1, 15]:
3
*** Code Execution Successful ***

```

Завдання 2.

```
using System;
```

```
class Program
```

```

{
    static void Main()
    {

```

```
Console.WriteLine("Enter the three sides of the triangle:");

double a = double.Parse(Console.ReadLine());

double b = double.Parse(Console.ReadLine());

double c = double.Parse(Console.ReadLine());


if (IsValidTriangle(a, b, c))
{
    double perimeter = a + b + c;

    double s = perimeter / 2;

    double area = Math.Sqrt(s * (s - a) * (s - b) * (s - c));

    Console.WriteLine($"Perimeter of the triangle: {perimeter}");

    Console.WriteLine($"Area of the triangle: {area}");

    Console.WriteLine($"Type of the triangle: {GetTriangleType(a, b, c)}");
}

else
{
    Console.WriteLine("This is not a valid triangle.");
}
}


static bool IsValidTriangle(double a, double b, double c)
```

```

{
    return a + b > c && a + c > b && b + c > a;
}

```

```
static string GetTriangleType(double a, double b, double c)
```

```

{
    if (a == b && b == c)
        return "Equilateral";

    else if (a == b || b == c || a == c)
        return "Isosceles";

    else
        return "Scalene";
}
}

```

The screenshot shows a C# IDE with a file named 'Main.cs'. The code defines a method `IsValidTriangle` and a static method `GetTriangleType`. The `GetTriangleType` method uses conditional logic to determine the type of a triangle based on the lengths of its three sides. The output window shows the results of running the program, indicating that the triangle is equilateral with a perimeter of 30 and an area of approximately 43.30.

```

Main.cs
20     Console.WriteLine($"Type of the triangle: {GetTriangleType(a, b, c
21     });
22     }
23     else
24     {
25         Console.WriteLine("This is not a valid triangle.");
26     }
27
28     static bool IsValidTriangle(double a, double b, double c)
29     {
30         return a + b > c && a + c > b && b + c > a;
31     }
32
33     static string GetTriangleType(double a, double b, double c)
34     {
35         if (a == b && b == c)
36             return "Equilateral";
37         else if (a == b || b == c || a == c)
38             return "Isosceles";
39         else
40             return "Scalene";
41     }
42 }
43
Output
mono /tmp/BHJGr2k50T.exe
Enter the three sides of the triangle:
10
100
10
Perimeter of the triangle: 30
Area of the triangle: 43.3012701892219
Type of the triangle: Equilateral
=== Code Execution Successful ===

```

Завдання 3.

```
using System;
```

```
using System.Linq;
```

```
class Program
```

```
{
```

```
    static void Main()
```

```
    {
```

```
        int lastDigit = 5;
```

```
        int length = 10 + lastDigit;
```

```
        int[] array = new int[length];
```

```
        Random rnd = new Random();
```

```
        for (int i = 0; i < length; i++)
```

```
        {
```

```
            array[i] = rnd.Next(-100, 101);
```

```
        }
```

```
        int min = array.Min();
```

```
        int max = array.Max();
```

```

        Console.WriteLine("Array:");

        foreach (int number in array)
        {
            Console.Write(number + " ");
        }

        Console.WriteLine();

        Console.WriteLine($"Minimum value: {min}");

        Console.WriteLine($"Maximum value: {max}");

    }
}

```

The screenshot shows a Visual Studio IDE with a C# file named 'Main.cs' and an 'Output' window. The code in 'Main.cs' is as follows:

```

1 using System;
2 using System.Linq;
3
4 class Program
5 {
6     static void Main()
7     {
8         int lastDigit = 5;
9         int length = 10 + lastDigit;
10        int[] array = new int[length];
11
12        Random rnd = new Random();
13        for (int i = 0; i < length; i++)
14        {
15            array[i] = rnd.Next(-100, 101);
16        }
17
18        int min = array.Min();
19        int max = array.Max();
20
21        Console.WriteLine("Array:");
22        foreach (int number in array)
23        {
24            Console.Write(number + " ");
25        }

```

The 'Output' window shows the following text:

```

- mono: /tmp/FHary2yrTS.exe
Array:
-91 28 91 -99 71 31 -33 -21 -73 18 -15 -87 72 11 -22
Minimum value: -99
Maximum value: 91

=== Code Execution Successful ===

```

Завдання 4.

```
using System;
```

```
using System.Linq;
```

```
class Program
```

```
{
```

```
    static void Main()
```

```
    {
```

```
        int lastDigit = 5;
```

```
        int length = 10 + lastDigit;
```

```
        int[] X = new int[length];
```

```
        Random rnd = new Random();
```

```
        for (int i = 0; i < length; i++)
```

```
        {
```

```
            X[i] = rnd.Next(-100, 101);
```

```
        }
```

```
        Console.WriteLine("Enter the number M:");
```

```
        int M = int.Parse(Console.ReadLine());
```

```
        int[] Y = X.Where(n => Math.Abs(n) > M).ToArray();
```

```
        Console.WriteLine("Array X:");
```

```
        foreach (int number in X)
```

```
        {
```

```

        Console.Write(number + " ");
    }

    Console.WriteLine();

    Console.WriteLine($"Number M: {M}");

    Console.WriteLine("Array Y:");
    foreach (int number in Y)
    {
        Console.Write(number + " ");
    }

    Console.WriteLine();
}
}

```

The screenshot shows a Visual Studio Code editor with a C# file named 'Main.cs'. The code is as follows:

```

16 }
17
18 Console.WriteLine("Enter the number M:");
19 int M = int.Parse(Console.ReadLine());
20
21 int[] Y = X.Where(n => Math.Abs(n) > M).ToArray();
22
23 Console.WriteLine("Array X:");
24 foreach (int number in X)
25 {
26     Console.Write(number + " ");
27 }
28 Console.WriteLine();
29
30 Console.WriteLine($"Number M: {M}");
31
32 Console.WriteLine("Array Y:");
33 foreach (int number in Y)
34 {
35     Console.Write(number + " ");
36 }
37 Console.WriteLine();
38 }
39 }
40

```

The 'Output' window on the right shows the execution results:

```

mono /tmp/SPcYB4EnQv.exe
Enter the number M:
5
Array X:
-6 -64 71 -70 -86 -61 -25 65 78 11 -61 64 4 -47 -89
Number M: 5
Array Y:
-6 -64 71 -70 -86 -61 -25 65 78 11 -61 64 -47 -89
=== Code Execution Successful ===

```


