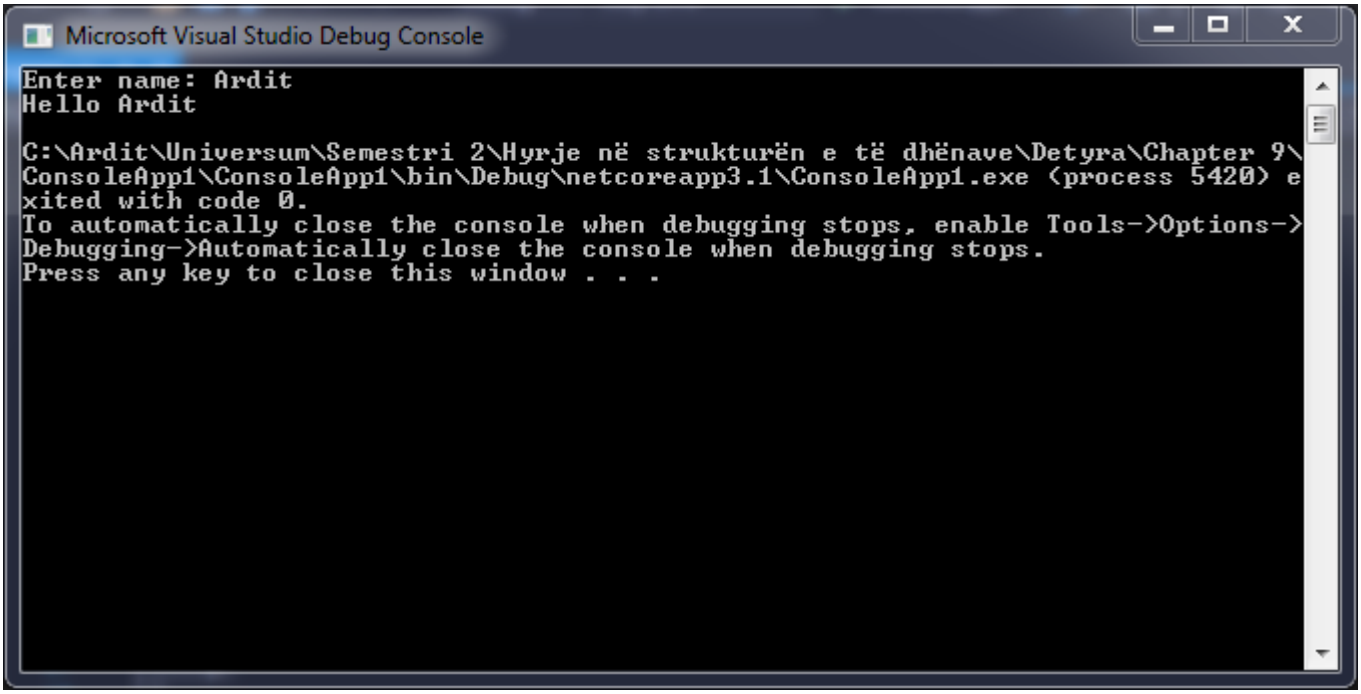
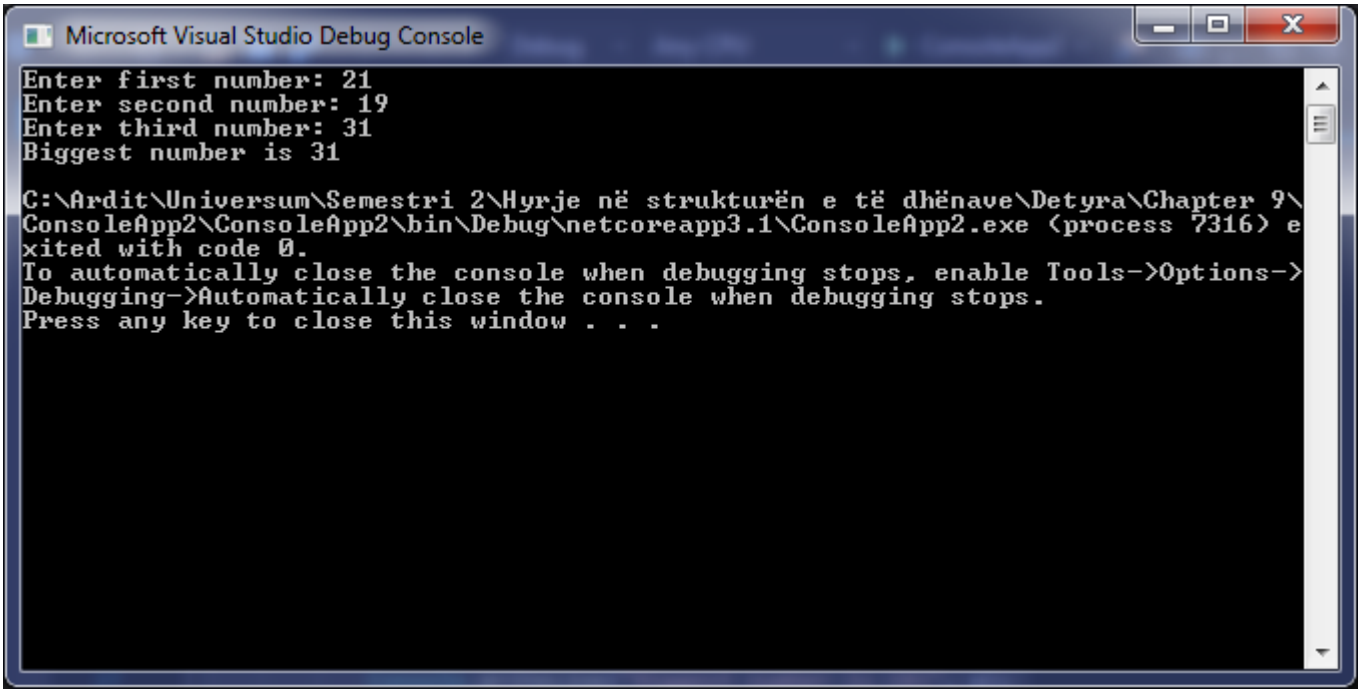
1. Write a method that by given name prints on the console `"Hello, !"` (for example: `"Hello, Peter!"`).

```
using System;

namespace ConsoleApp1
{
    class Program
    {
        static void ReturnName(string name)
        {
            Console.WriteLine("Hello {0}", name);
        }

        static void Main(string[] args)
        {
            Console.Write("Enter name: ");
            ReturnName(Console.ReadLine());
        }
    }

}
```



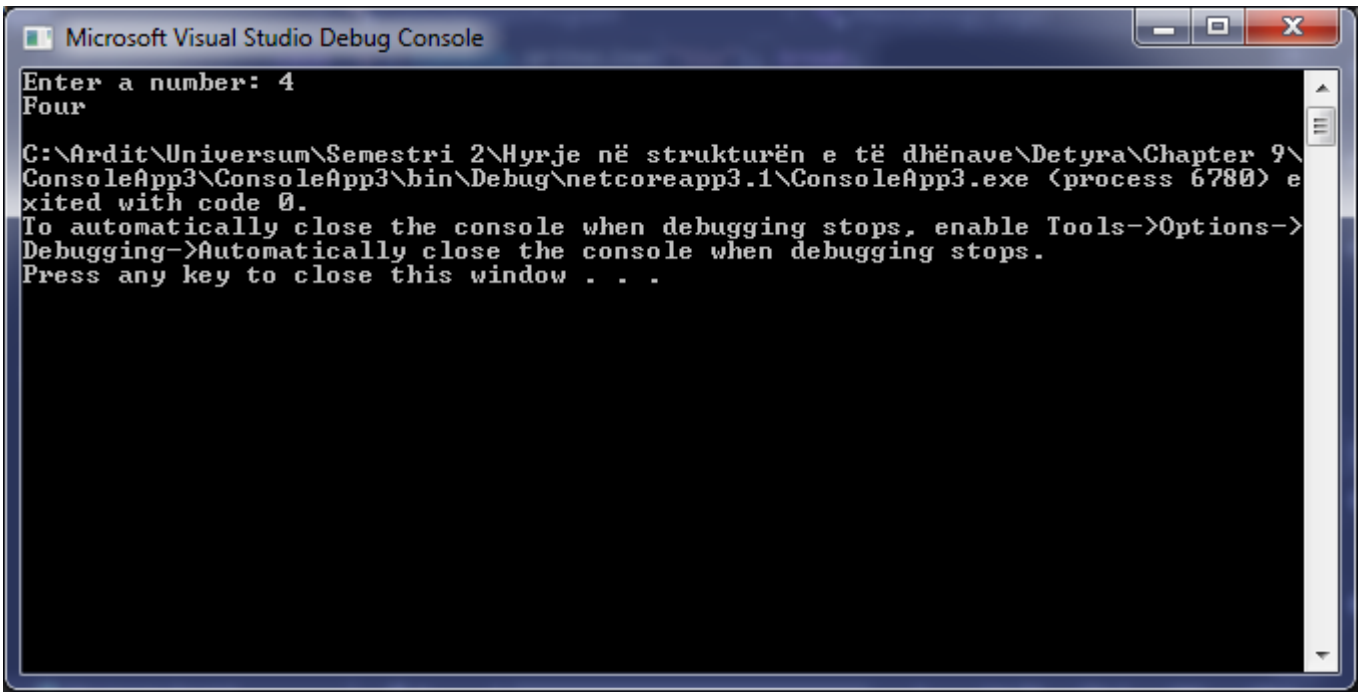
2. Create a method `GetMax()` with two integer (`int`) parameters, that returns maximal of the two numbers. Write a program that reads three numbers from the console and prints the biggest of them. Use the `GetMax()` method you just created.

```
using System;

namespace ConsoleApp2
{
    class Program
    {
        static int a;

        static void GetMax(int first, int second)
        {
            if (first > second) a = first;
            else a = second;
        }

        static void Main(string[] args)
        {
            Console.Write("Enter first number: ");
            a = Int32.Parse(Console.ReadLine());
            Console.Write("Enter second number: ");
            int b = Int32.Parse(Console.ReadLine());
            Console.Write("Enter third number: ");
            int c = Int32.Parse(Console.ReadLine());

            GetMax(a, b);
            GetMax(a, c);

            Console.WriteLine("Biggest number is {0}", a);
        }
    }
}
```



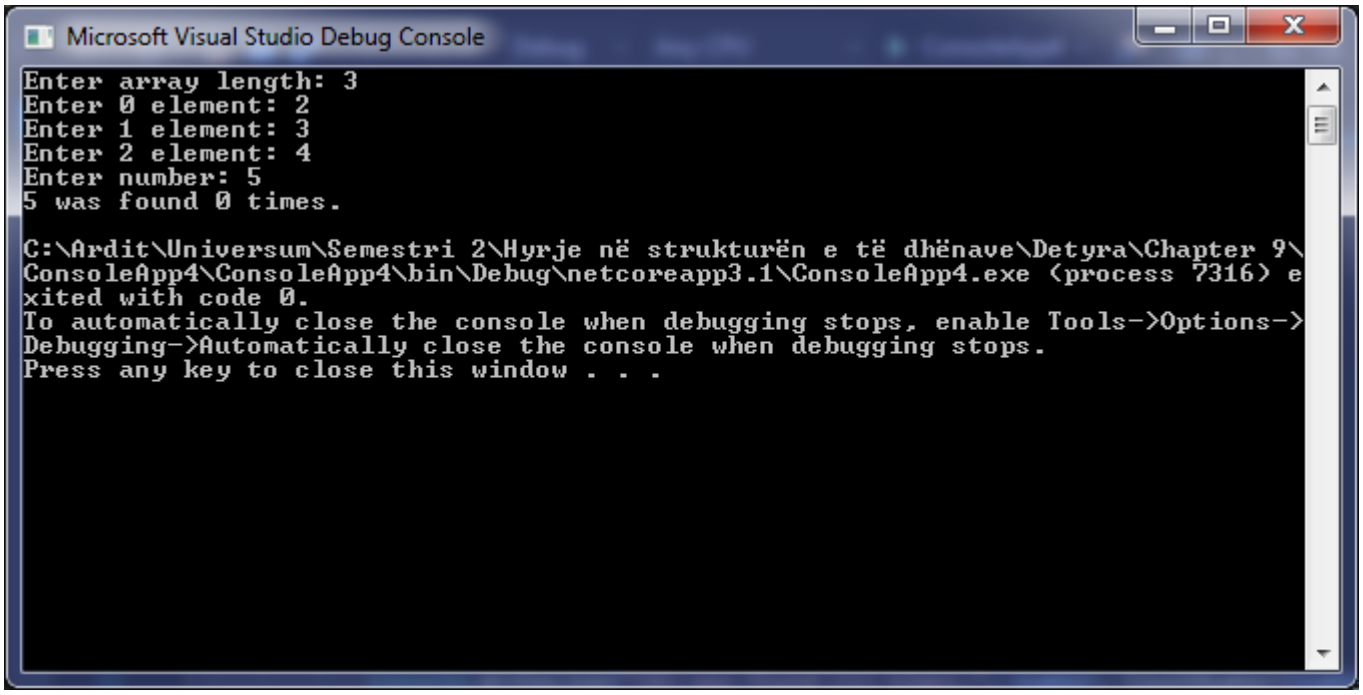
3. Write a method that returns **the English name of the last digit** of a given number. Example: for 512 prints "two"; for 1024 prints "four"

```
using System;

namespace ConsoleApp3
{
    class Program
    {
        static void GetName(string number)
        {
            switch (number[number.Length - 1])
            {
                case '1': Console.WriteLine("One"); break;
                case '2': Console.WriteLine("Two"); break;
                case '3': Console.WriteLine("Three"); break;
                case '4': Console.WriteLine("Four"); break;
                case '5': Console.WriteLine("Five"); break;
                case '6': Console.WriteLine("Six"); break;
                case '7': Console.WriteLine("Seven"); break;
                case '8': Console.WriteLine("Eight"); break;
                case '9': Console.WriteLine("Nine"); break;
                case '0': Console.WriteLine("Zero"); break;
                default: Console.WriteLine("Wrong input."); break;
            }
        }

        static void Main(string[] args)
        {
            Console.Write("Enter a number: ");
            string number = Console.ReadLine();

            GetName(number);
        }
    }
}
```



4. Write a method that finds **how many times certain number can be found in a given array**.

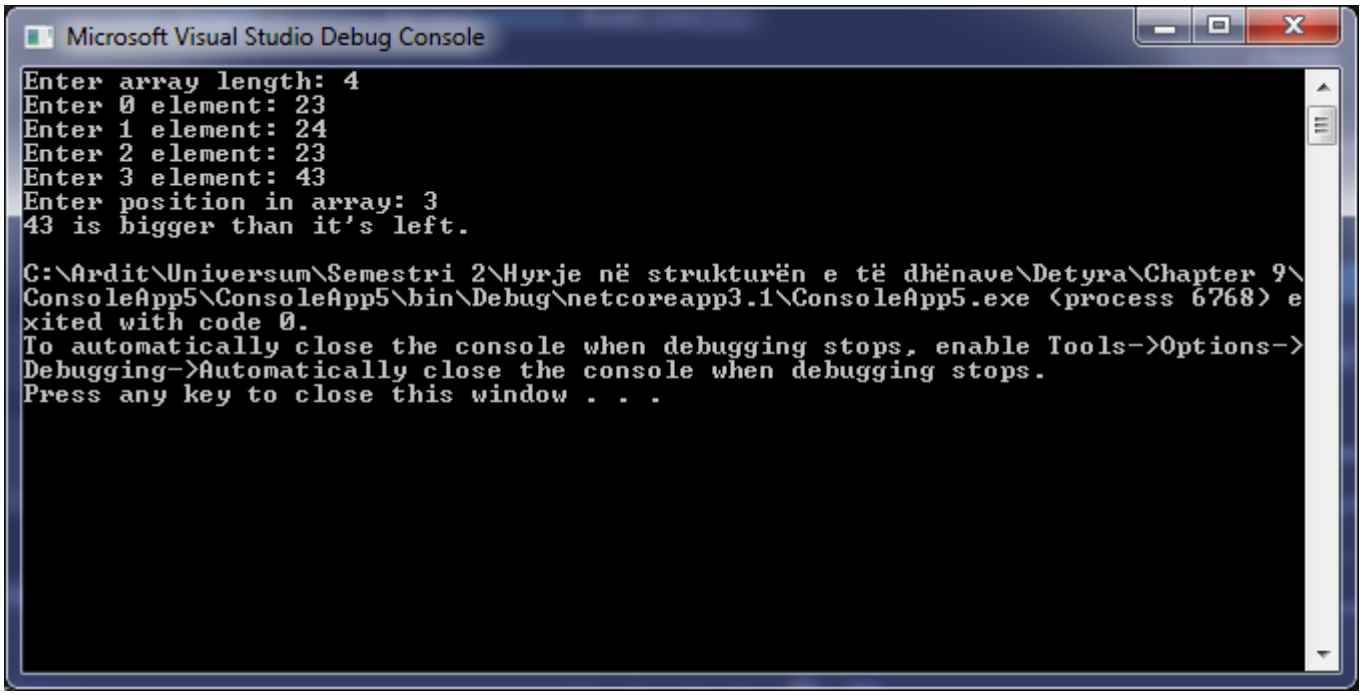
```
using System;

namespace ConsoleApp4
{
    class Program
    {
        static int CountNumber(int number, int[] arr)
        {
            int counter = 0;

            for (int i = 0; i < arr.Length; i++) if (number == arr[i]) counter++;

            return counter;
        }

        static void Main(string[] args)
        {
            Console.Write("Enter array length: ");
            int length = Int32.Parse(Console.ReadLine());

            int[] arr = new int[length];

            for (int i = 0; i < arr.Length; i++)
            {
                Console.Write("Enter {0} element: ", i);
                arr[i] = Int32.Parse(Console.ReadLine());
            }

            Console.Write("Enter number: ");
            int number = Int32.Parse(Console.ReadLine());

            Console.WriteLine("{0} was found {1} times.", number, CountNumber(number, arr));
        }
    }
}
```



5. Write a method that checks whether an element, from a certain position in an array is **greater than its two neighbours**.

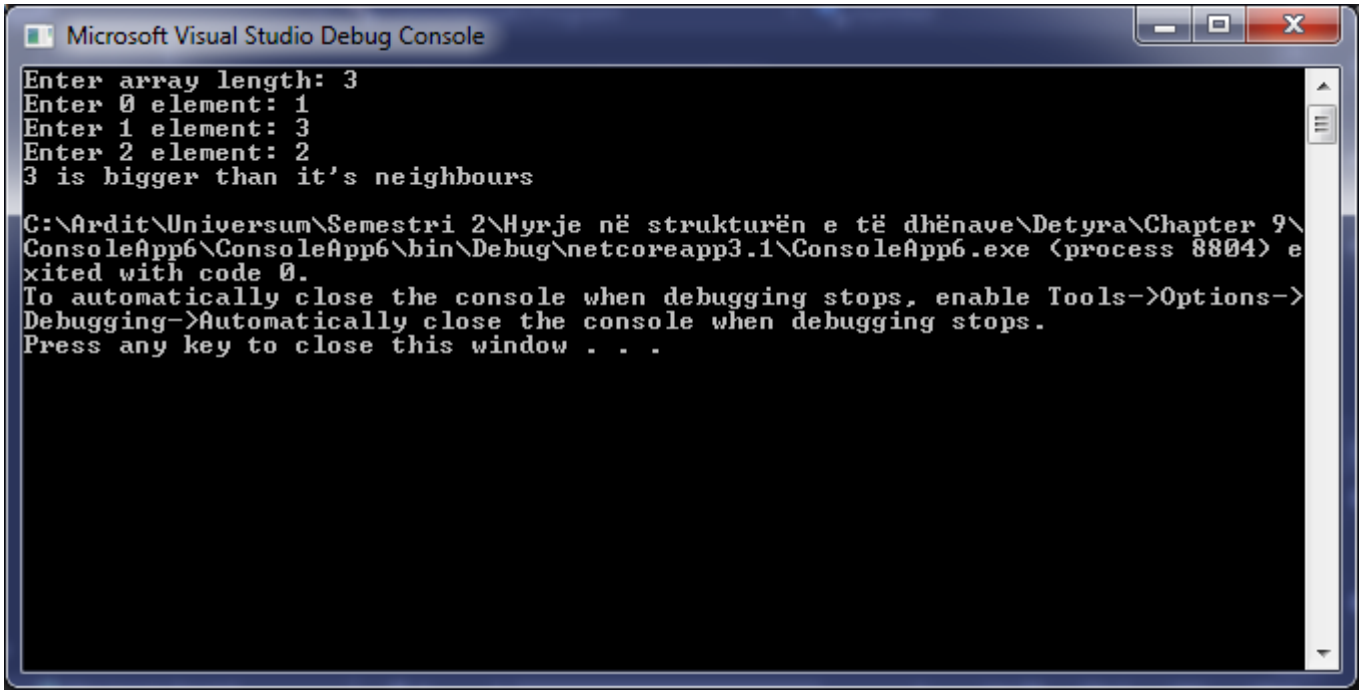
```
using System;

namespace ConsoleApp5
{
    class Program
    {
        static void CompareNumber(int pos, int[] arr)
        {
            if (pos == 0)
            {
                if (arr[0] < arr[1]) Console.WriteLine("{0} is smaller than it's right.", arr[0]);
                else if (arr[0] > arr[1]) Console.WriteLine("{0} is bigger than it's right.", arr[0]);
                else Console.WriteLine("{0} is equal to it's right.", arr[0]);
            }
            else if (pos == arr.Length - 1)
            {
                if (arr[arr.Length - 1] < arr[arr.Length - 2]) Console.WriteLine("{0} is smaller than
it's left.", arr[pos]);
                else if (arr[arr.Length - 1] > arr[arr.Length - 2]) Console.WriteLine("{0} is bigger
than it's left.", arr[pos]);
                else Console.WriteLine("{0} is equal to it's left.", arr[pos]);
            }
            else
            {
                if (arr[pos] < arr[pos - 1])
                {
                    if (arr[pos] < arr[pos + 1]) Console.WriteLine("{0} is smaller than it's
neighbours.", arr[pos]);
                    else if (arr[pos] == arr[pos + 1]) Console.WriteLine("{0} is smaller than it's left
and equal to it's right.", arr[pos]);
                    else Console.WriteLine("{0} is smaller than it's left and bigger than it's right.",
arr[pos]);
                }
                else if (arr[pos] == arr[pos - 1])
                {
                    if (arr[pos] < arr[pos + 1]) Console.WriteLine("{0} is euqal to it's left and
smaller than it's right.", arr[pos]);
                    else if (arr[pos] == arr[pos + 1]) Console.WriteLine("{0} is equal to it's
neighbours.", arr[pos]);
                    else Console.WriteLine("{0} is equal to it's left and bigger than it's right.",
arr[pos]);
                }
                else
                {
                    if (arr[pos] < arr[pos + 1]) Console.WriteLine("{0} is bigger than it's left and
smaller than it's right.", arr[pos]);
                    else if (arr[pos] == arr[pos + 1]) Console.WriteLine("{0} is bigger than it's left
and equal to it's right.", arr[pos]);
                    else Console.WriteLine("{0} is bigger than it's neighbours.", arr[pos]);
                }
            }
        }

        static void Main(string[] args)
        {
            Console.Write("Enter array length: ");
            int length = Int32.Parse(Console.ReadLine());

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

```
            for (int i = 0; i < arr.Length; i++)
            {
                Console.Write("Enter {0} element: ", i);
                arr[i] = Int32.Parse(Console.ReadLine());
            }

            Console.Write("Enter position in array: ");
            int pos = Int32.Parse(Console.ReadLine());

            CompareNumber(pos, arr);
        }
    }
}
```



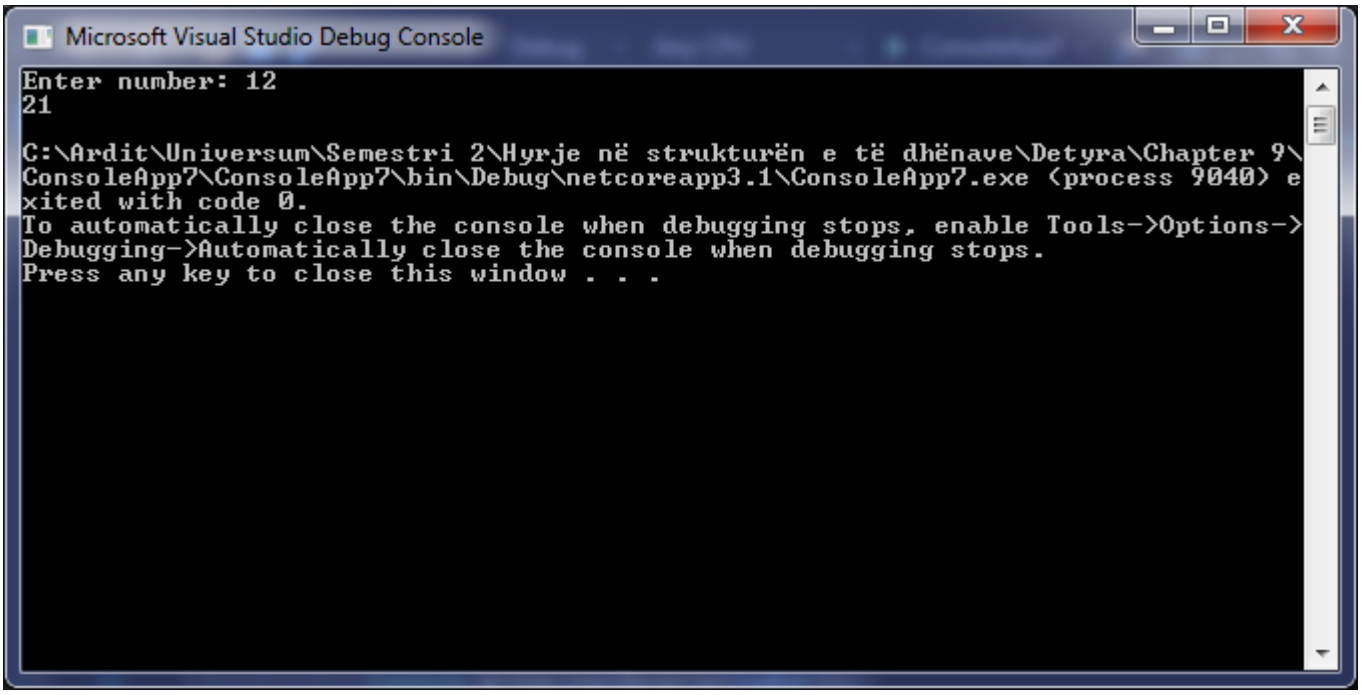
6. Write a method that returns the position of **the first occurrence** of an element from an array, such that it is greater than its two neighbors simultaneously. Otherwise the result must be -1.

```
using System;

namespace ConsoleApp6
{
    class Program
    {
        static int number = int.MinValue;

        static void CompareNumber(int[] arr)
        {
            for (int i = 1; i < arr.Length - 1; i++)
                if (arr[i] > arr[i - 1] && arr[i] > arr[i + 1])
                {
                    number = arr[i];
                    break;
                }
        }

        static void Main(string[] args)
        {
            Console.Write("Enter array length: ");
            int length = Int32.Parse(Console.ReadLine());

            int[] arr = new int[length];

            for (int i = 0; i < arr.Length; i++)
            {
                Console.Write("Enter {0} element: ", i);
                arr[i] = Int32.Parse(Console.ReadLine());
            }

            CompareNumber(arr);

            if (number == int.MinValue) Console.WriteLine("-1");
            else Console.WriteLine("{0} is bigger than it's neighbours", number);
        }
    }
}
```


Microsoft Visual Studio Debug Console
Enter array length: 3
Enter 0 element: 1
Enter 1 element: 3
Enter 2 element: 2
3 is bigger than it's neighbours
C:\Ardit\Universum\Semestri 2\Hyrje në strukturën e të dhënave\Detyra\Chapter 9\ConsoleApp6\ConsoleApp6\bin\Debug\netcoreapp3.1\ConsoleApp6.exe (process 8804) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .

7. Write a method that prints the digits of a given decimal number in a reversed order. For example 256, must be printed as 652.

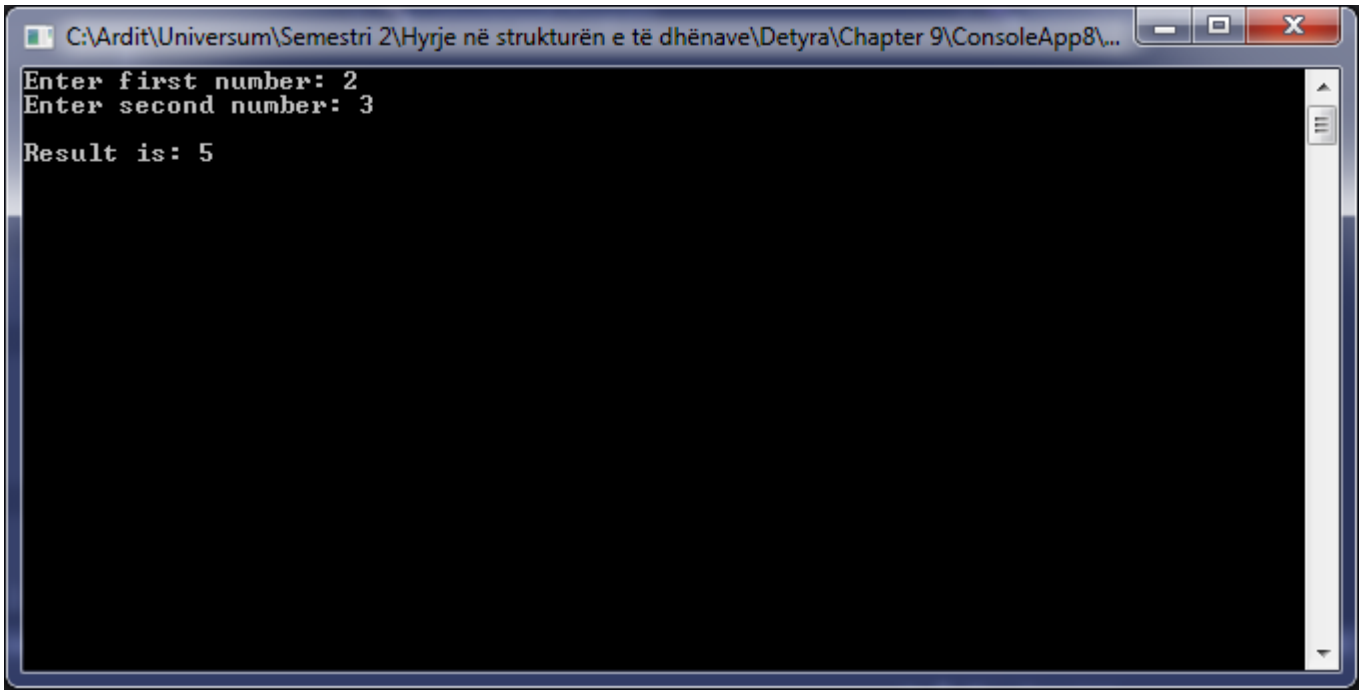
```
using System;

namespace ConsoleApp7
{
    class Program
    {
        public static string Reverse(string number)
        {
            char[] charArray = number.ToCharArray();
            Array.Reverse(charArray);
            return new string(charArray);
        }

        static void Main(string[] args)
        {
            Console.Write("Enter number: ");
            string number = Console.ReadLine();

            Console.WriteLine(Reverse(number));
        }
    }
}
```



8. Write a method that calculates the **sum of two very long positive integer numbers**. The numbers are represented as **array digits** and the last digit (the ones) is stored in the array at index 0. Make the method work for all numbers with length up to 10,000 digits.

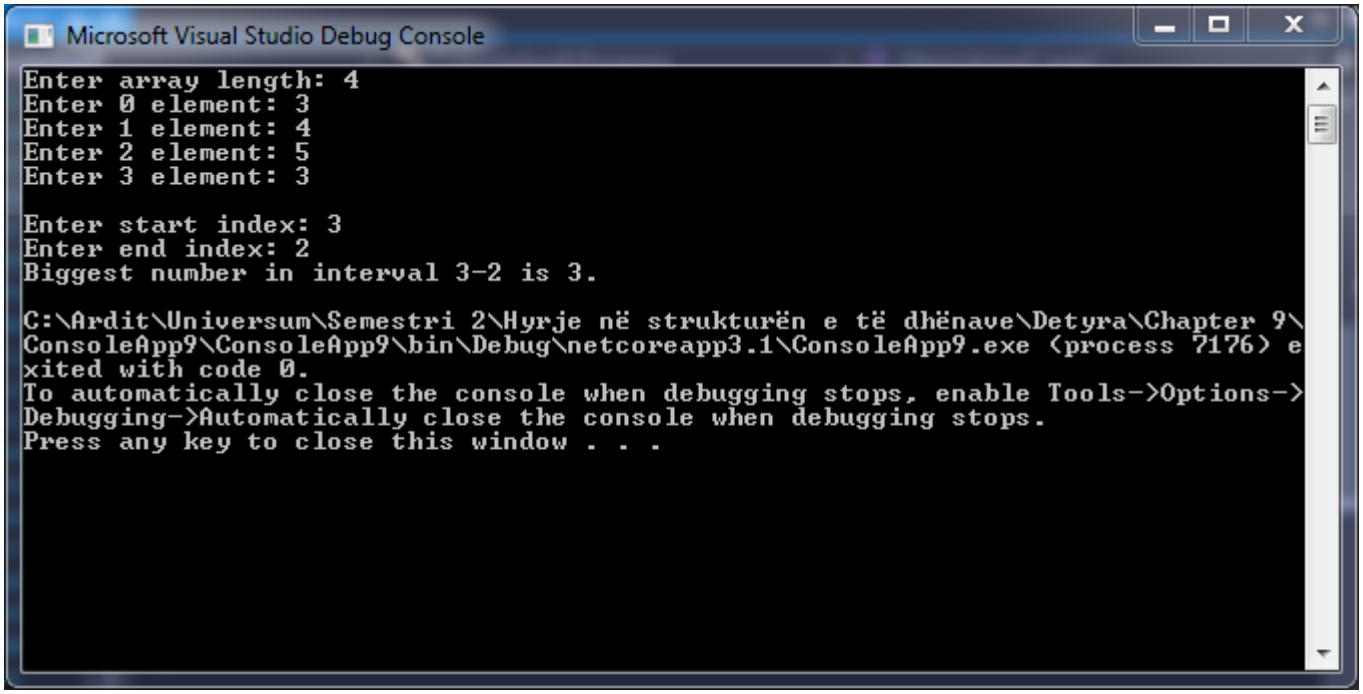
```
using System;

namespace ConsoleApp8
{
    class Program
    {
        static void Main(string[] args)
        {
            int length;
            int[] Array1 = new int[10000];
            int[] Array2 = new int[10000];

            Console.Write("Enter first number: ");
            String s1 = Console.ReadLine();
            length = s1.Length;

            for (int i = 0; i < s1.Length; i++)
            {
                Array1[i] = Convert.ToInt32(s1.Substring(s1.Length - 1 - i, 1));
            }

            Console.Write("Enter second number: ");
            String s2 = Console.ReadLine();

            if (s2.Length > length)
            {
                length = s2.Length;
            }

            for (int i = 0; i < s2.Length; i++)
            {
                Array2[i] = Convert.ToInt32(s2.Substring(s2.Length - 1 - i, 1));
            }

            for (int i = 0; i < length; i++)
            {
                Array1[i] += Array2[i];

                if (Array1[i] >= 10)
                {
                    Array1[i] -= 10;
                    Array1[i + 1]++;
                }
            }

            if (Array1[length] == 1)
            {
                length++;
            }

            Console.WriteLine();
            Console.Write("Result is: ");

            for (int i = 0; i < length; i++)
            {
                Console.Write(Array1[length - 1 - i]);
            }
```

```
            Console.ReadLine();
        }
    }
}
```



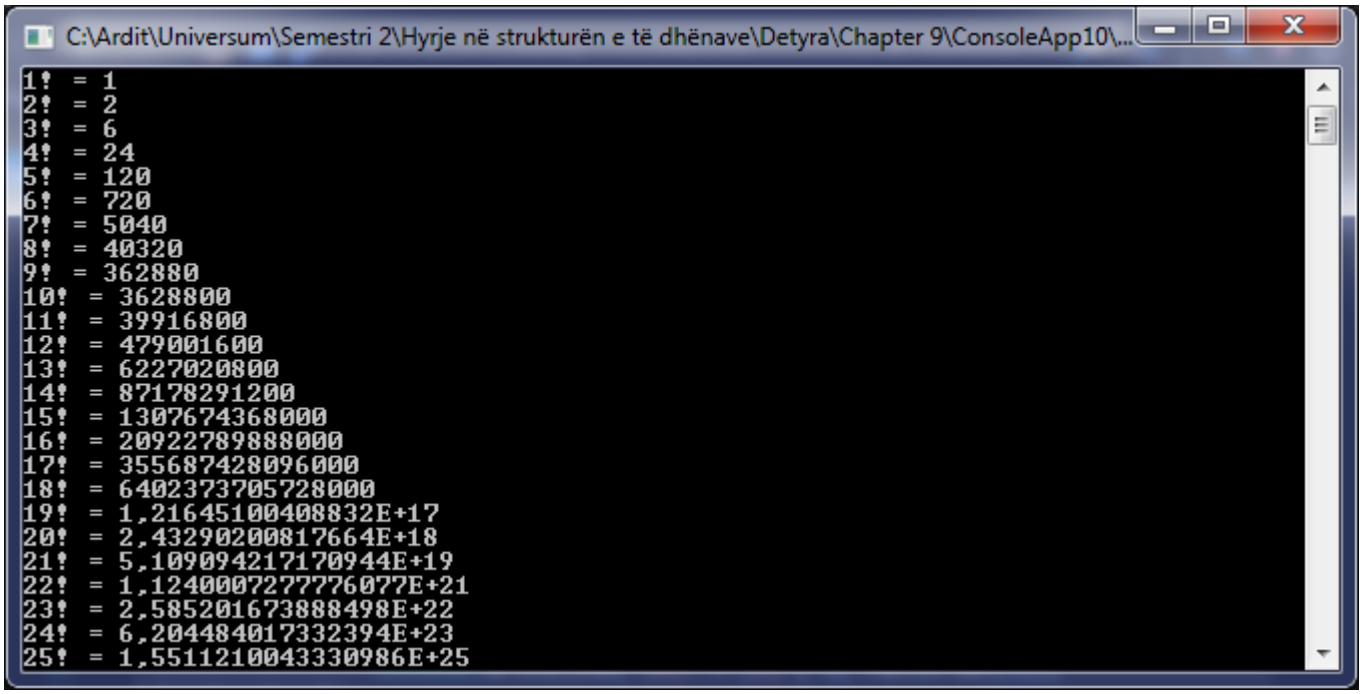
9. Write a method that finds the biggest element of an array. Use that method to implement sorting in descending order.

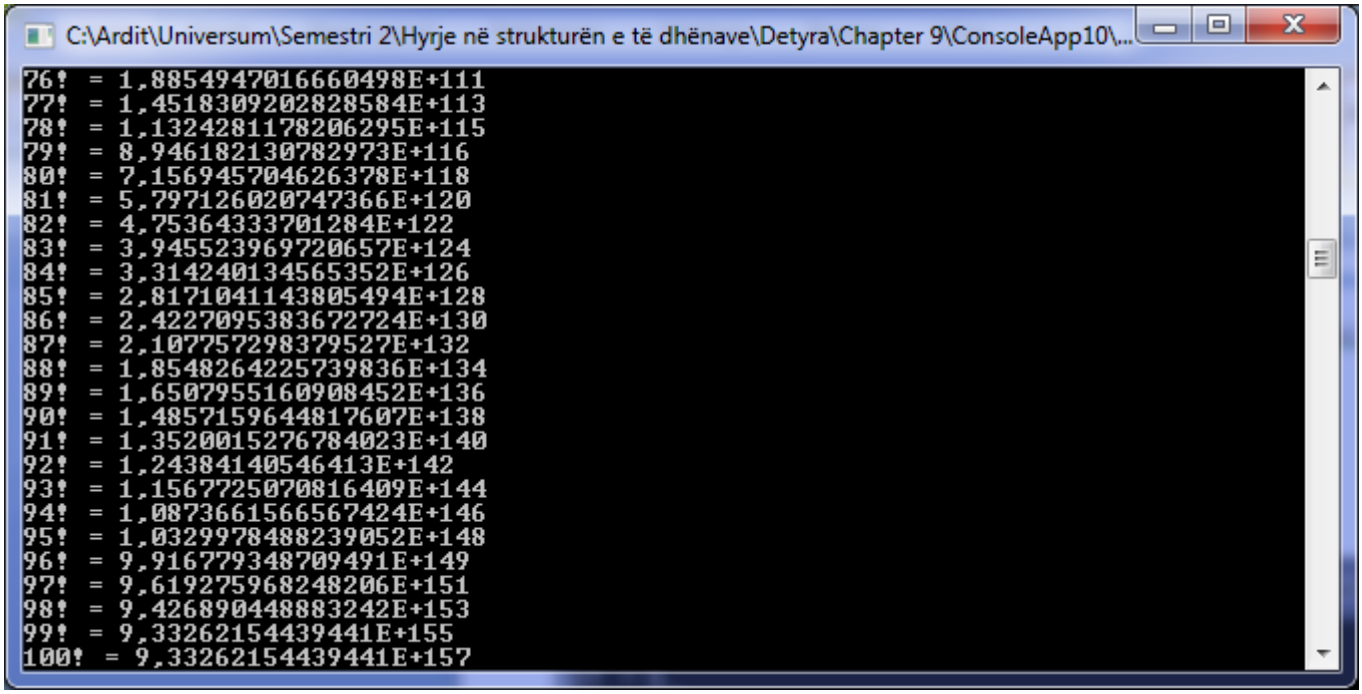
```
using System;

namespace ConsoleApp9
{
    class Program
    {
        public static int GetMax(int[] array, int start, int end)
        {
            int maxNum = array[start];

            for (int i = start + 1; i < end; i++)
            {
                if (array[i] > maxNum) maxNum = array[i];
            }

            return maxNum;
        }

        public static void Main(string[] args)
        {
            Console.Write("Enter array length: ");
            int length = Int32.Parse(Console.ReadLine());

            int[] arr = new int[length];

            for (int i = 0; i < arr.Length; i++)
            {
                Console.Write("Enter {0} element: ", i);
                arr[i] = Int32.Parse(Console.ReadLine());
            }

            Console.Write("\nEnter start index: ");
            int startIndex = Int32.Parse(Console.ReadLine());

            Console.Write("Enter end index: ");
            int endIndex = Int32.Parse(Console.ReadLine());

            Console.WriteLine("Biggest number in interval {0}-{1} is {2}.", startIndex, endIndex,
GetMax(arr, startIndex, endIndex));
        }
    }
}
```


Microsoft Visual Studio Debug Console
Enter array length: 4
Enter 0 element: 3
Enter 1 element: 4
Enter 2 element: 5
Enter 3 element: 3
Enter start index: 3
Enter end index: 2
Biggest number in interval 3-2 is 3.
C:\Ardit\Universum\Semestri 2\Hyrje në strukturën e të dhënave\Detyra\Chapter 9\ConsoleApp9\ConsoleApp9\bin\Debug\netcoreapp3.1\ConsoleApp9.exe (process 7176) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .

10. Write a program that calculates and prints the `n!` for any n in the range `[1…100]` .

```
using System;

namespace ConsoleApp10
{
    class Program
    {
        static double Factorial(double number)
        {
            if (number <= 1)
                return 1;
            else
                return number * Factorial(number - 1);
        }

        static void Main(string[] args)
        {
            for (int i = 1; i < 101; i++)
            {
                Console.WriteLine("{0}! = {1}", i, Factorial(i));
            }
            Console.ReadLine();
        }
    }
}
```

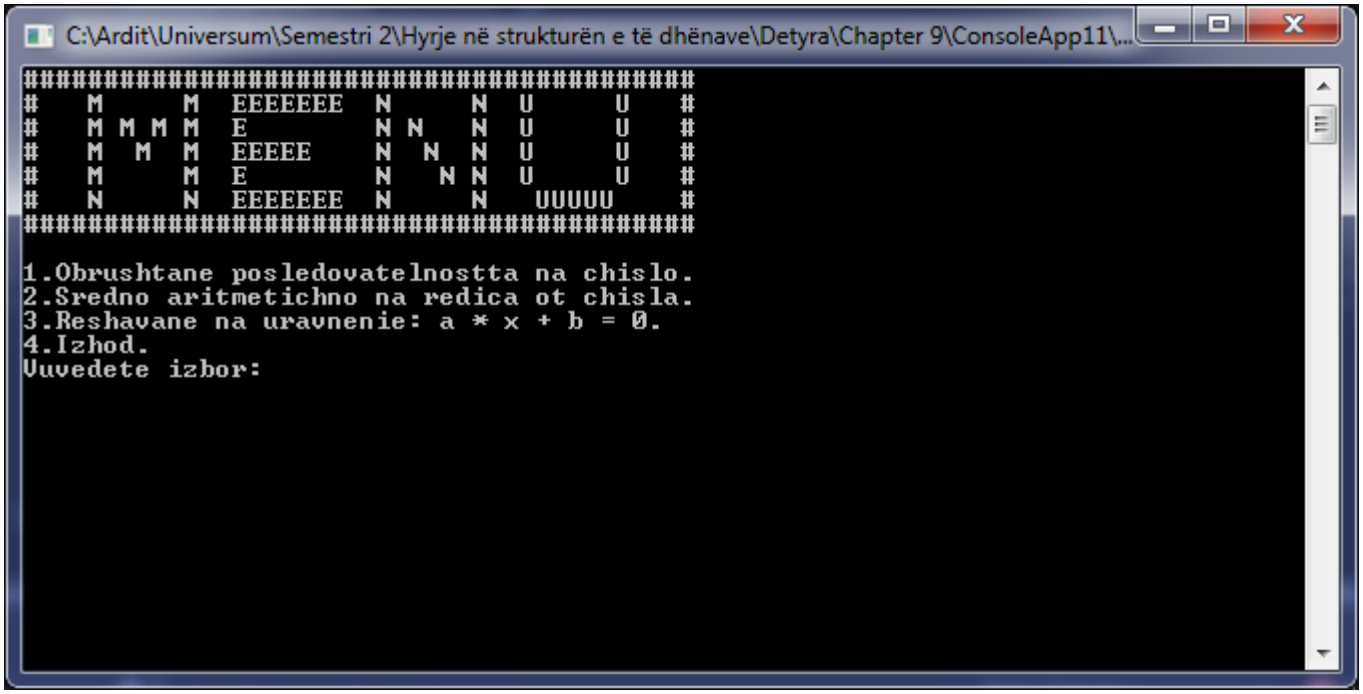



C:\Ardit\Universum\Semestri 2\Hyrje në strukturën e të dhënave\Detyra\Chapter 9\ConsoleApp10\...
76! = 1,88549470166604 98E+111
77! = 1,45183092028285 84E+113
78! = 1,13242811782062 95E+115
79! = 8,94618213078297 3E+116
80! = 7,15694570462637 8E+118
81! = 5,79712602074736 6E+120
82! = 4,75364333701284E+122
83! = 3,94552396972065 7E+124
84! = 3,31424013456535 2E+126
85! = 2,81710411438054 94E+128
86! = 2,42270953836727 24E+130
87! = 2,10775729837952 7E+132
88! = 1,85482642257398 36E+134
89! = 1,65079551609084 52E+136
90! = 1,48571596448176 07E+138
91! = 1,35200152767840 23E+140
92! = 1,2438414054641 3E+142
93! = 1,15677250708164 09E+144
94! = 1,08736615665674 24E+146
95! = 1,03299784882390 52E+148
96! = 9,91677934870949 1E+149
97! = 9,61927596824820 6E+151
98! = 9,42689044888324 2E+153
99! = 9,33262154439441E+155
100! = 9,33262154439441E+157

11. Write a program that solves the following tasks:
- Put the digits from an integer number into a reversed order.
- Calculate the average of given sequence of numbers.
- Solve the linear equation `a * x + b = 0`.

```
using System;

namespace ConsoleApp11
{
    class Program
    {

        public static string ReverseString(string s)
        {
            char[] arr = s.ToCharArray();
            Array.Reverse(arr);
            return new string(arr);
        }

        static void Reverse()
        {
            int numberReverse;
            string stringNumberReverse;
            do
            {
                Console.Clear();
                Console.Write("Vuvedete neotricatelno chislo: ");
                stringNumberReverse = Console.ReadLine();
                numberReverse = int.Parse(stringNumberReverse);
            } while (numberReverse < 0);

            Console.WriteLine("Chisloto oburnato " + ReverseString(stringNumberReverse));
            Console.ReadLine();
        }

        static void Average()
        {
            int numberAverage = 0;
            int entries = -1;
            int temp;
            string numberAverageString;
            bool input;
            Console.Clear();

            do
            {
                Console.Write("Vuvedete chislo ot redicata. Vuvedete bukva za da prikluchite: ");
                numberAverageString = Console.ReadLine();
                input = Int32.TryParse(numberAverageString, out temp);
                numberAverage += temp;
                entries++;
            } while (input);

            Console.WriteLine("Srednoto aritmetichno e {0}.", (float)numberAverage / entries);
            Console.ReadLine();
        }

        static void SolveEquation()
        {
            int a = 0;

            do
```

```
            {
                Console.Clear();
                Console.Write("Vuvedete a: ");
                a = int.Parse(Console.ReadLine());
            } while (a == 0);

            Console.Write("Vuvedete b: ");
            int b = int.Parse(Console.ReadLine());

            Console.WriteLine("x = {0}", (float)-b / a);
            Console.ReadLine();
        }

        static void Main(string[] args)
        {
            byte choice;

            do
            {
                Console.Clear();
                Console.WriteLine("##############################################");
                Console.WriteLine("#   M     M  EEEEEEE  N     N  U     U   #");
                Console.WriteLine("#   M M M M  E        N N   N  U     U   #");
                Console.WriteLine("#   M  M  M  EEEEE    N  N  N  U     U   #");
                Console.WriteLine("#   M     M  E        N   N N  U     U   #");
                Console.WriteLine("#   N     N  EEEEEEE  N     N   UUUUU    #");
                Console.WriteLine("##############################################" +
System.Environment.NewLine);
                Console.WriteLine("1.Obrushtane posledovatelnostta na chislo.");
                Console.WriteLine("2.Sredno aritmetichno na redica ot chisla.");
                Console.WriteLine("3.Reshavane na uravnenie: a * x + b = 0.");
                Console.WriteLine("4.Izhod.");
                Console.Write("Vuvedete izbor: ");

                choice = byte.Parse(Console.ReadLine());

                switch (choice)
                {
                    case 1: Reverse(); break;
                    case 2: Average(); break;
                    case 3: SolveEquation(); break;
                }

            } while (choice != 4);

            Console.WriteLine(System.Environment.NewLine + "Krai!");
            Console.ReadLine();
        }
    }
}
```


C:\Ardit\Universum\Semestri 2\Hyrje në strukturën e të dhënave\Detyra\Chapter 9\ConsoleApp11\...
MENU
1.Obrushtane posledovatelnostta na chislo.
2.Sredno aritmetichno na redica ot chisla.
3.Reshavane na uravnenie: a * x + b = 0.
4.Izhod.
Vuvedete izbor: