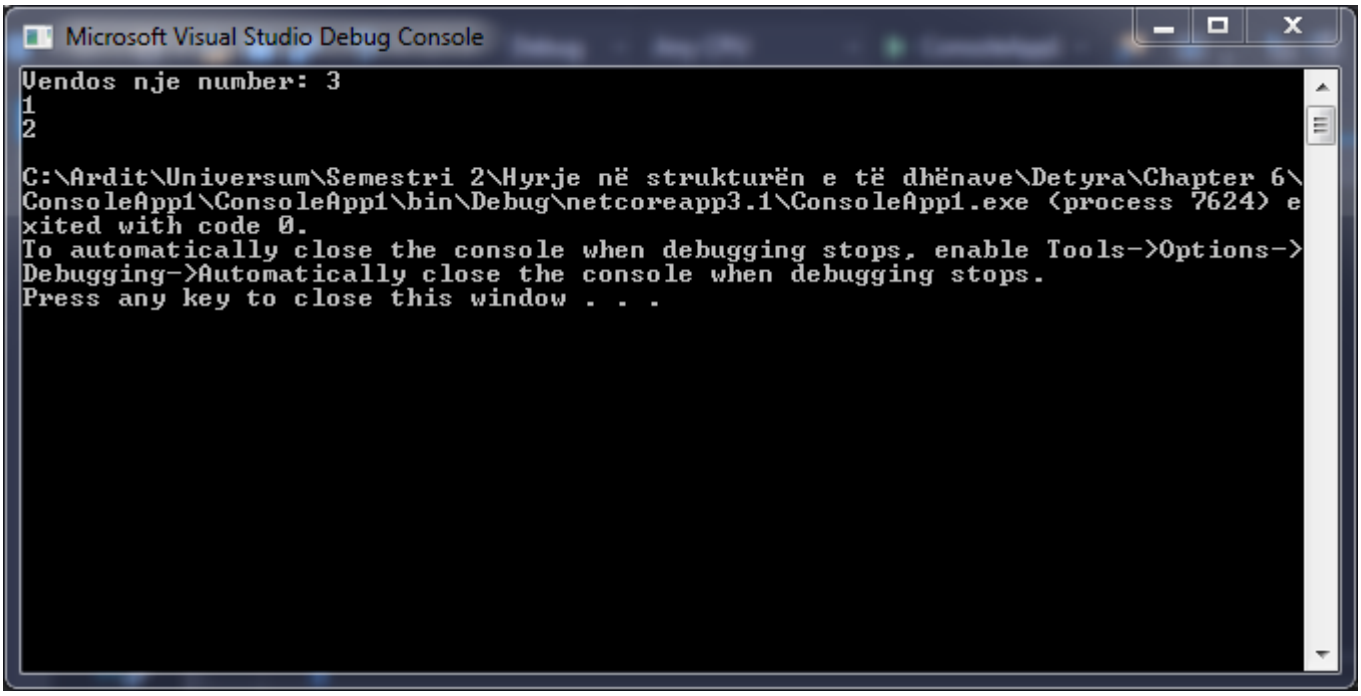
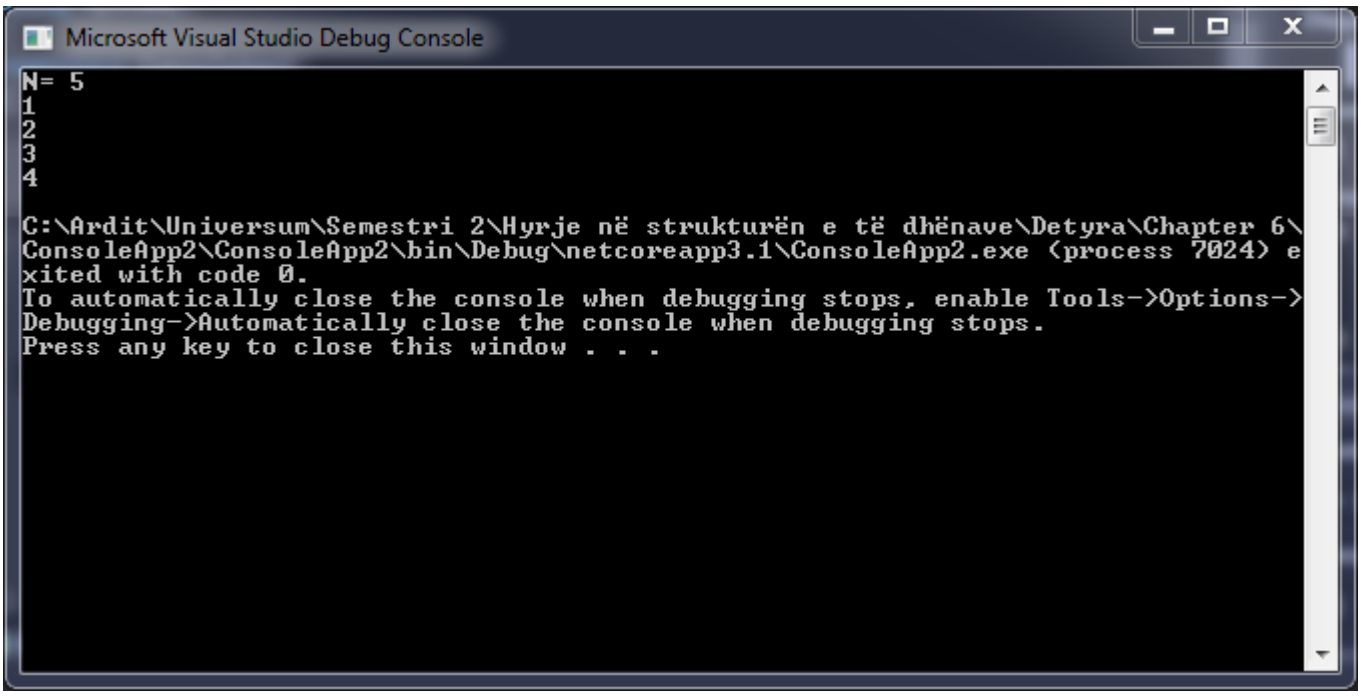
1. Write a program that prints on the console **the numbers from 1 to N**. The number **N** should be read from the standard input.

```
using System;

namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.Write("Vendos nje number: ");
            int length = Int32.Parse(Console.ReadLine());

            for (int i = 1; i < length; i++)
                Console.WriteLine(i);
        }
    }
}
```



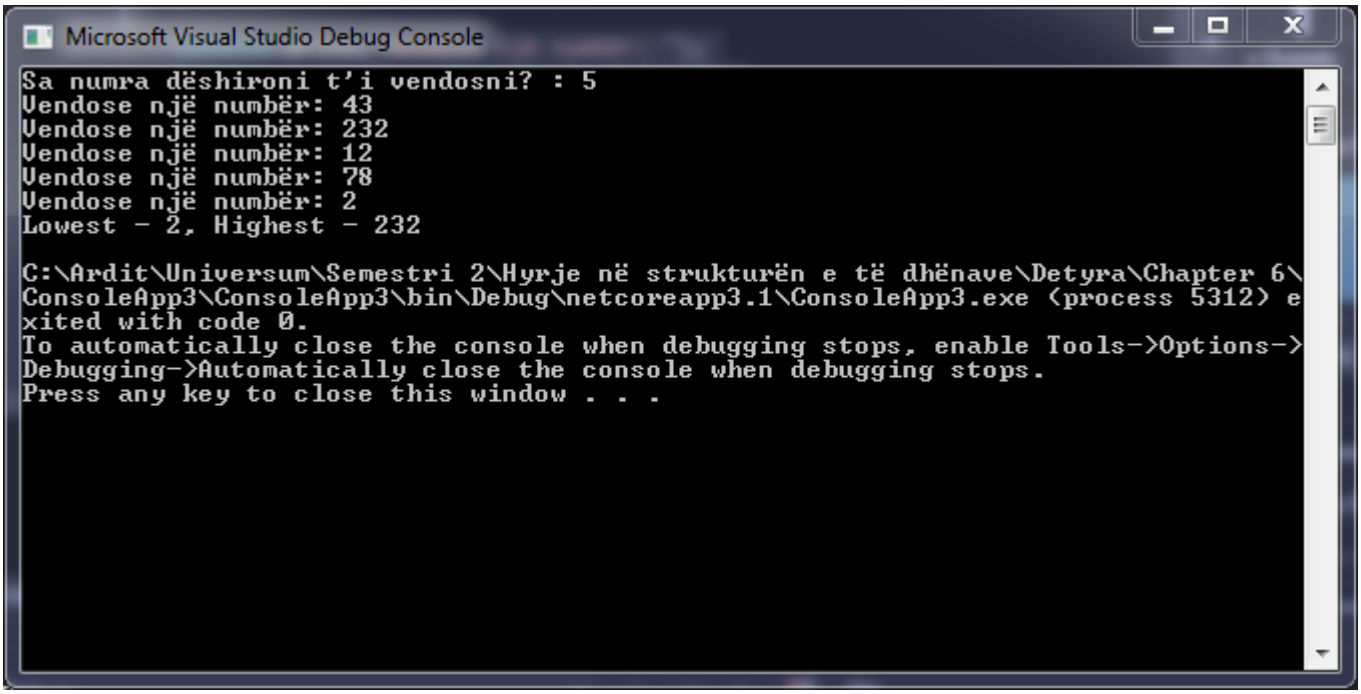
2. Write a program that prints on the console the numbers from 1 to N, which are **not divisible by 3 and 7 simultaneously**. The number N should be read from the standard input.

```
using System;

namespace ConsoleApp2
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.Write("N= ");
            int length = Int32.Parse(Console.ReadLine());

            for (int i = 1; i < length; i++)
            {
                if (i % (3 * 7) != 0) Console.WriteLine(i);
            }
        }
    }
}
```



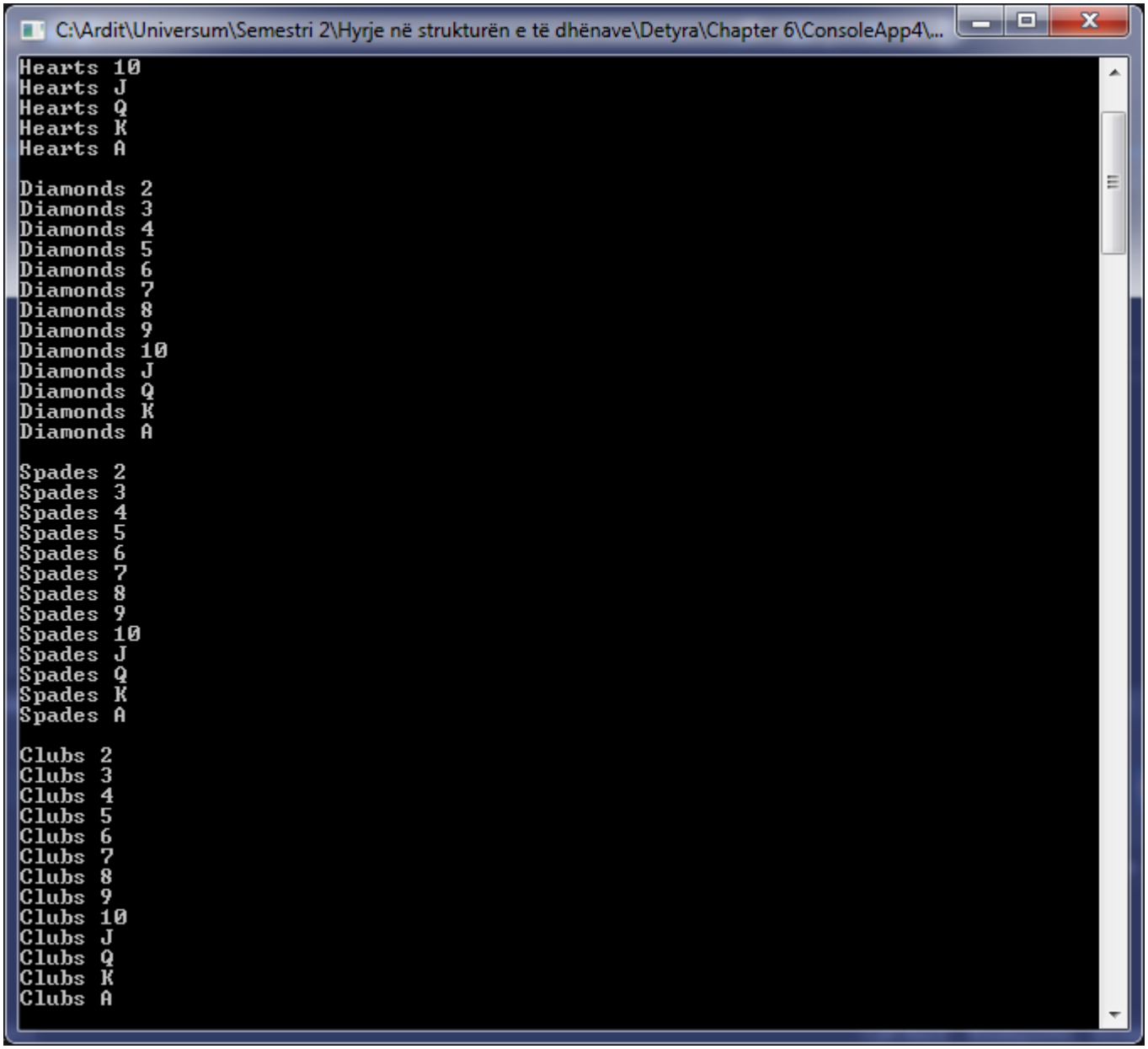
3. Write a program that reads from the console a series of integers and prints the **smallest** and **largest** of them.

```
using System;

namespace ConsoleApp3
{
    class Program
    {
        static void Main(string[] args)
        {
            int lowest = 0, highest = 0, input;

            Console.Write("Sa numra dëshironi t'i vendosni? : ");
            int lenght = Int32.Parse(Console.ReadLine());

            for (int i = 0; i < lenght; i++)
            {
                Console.Write("Vendose një numër: ");
                input = Int32.Parse(Console.ReadLine());
                if (i == 0) lowest = highest = input;
                else
                {
                    if (lowest > input) lowest = input;
                    if (highest < input) highest = input;
                }
            }
            Console.WriteLine("Lowest - {0}, Highest - {1}", lowest, highest);
        }
    }
}
```



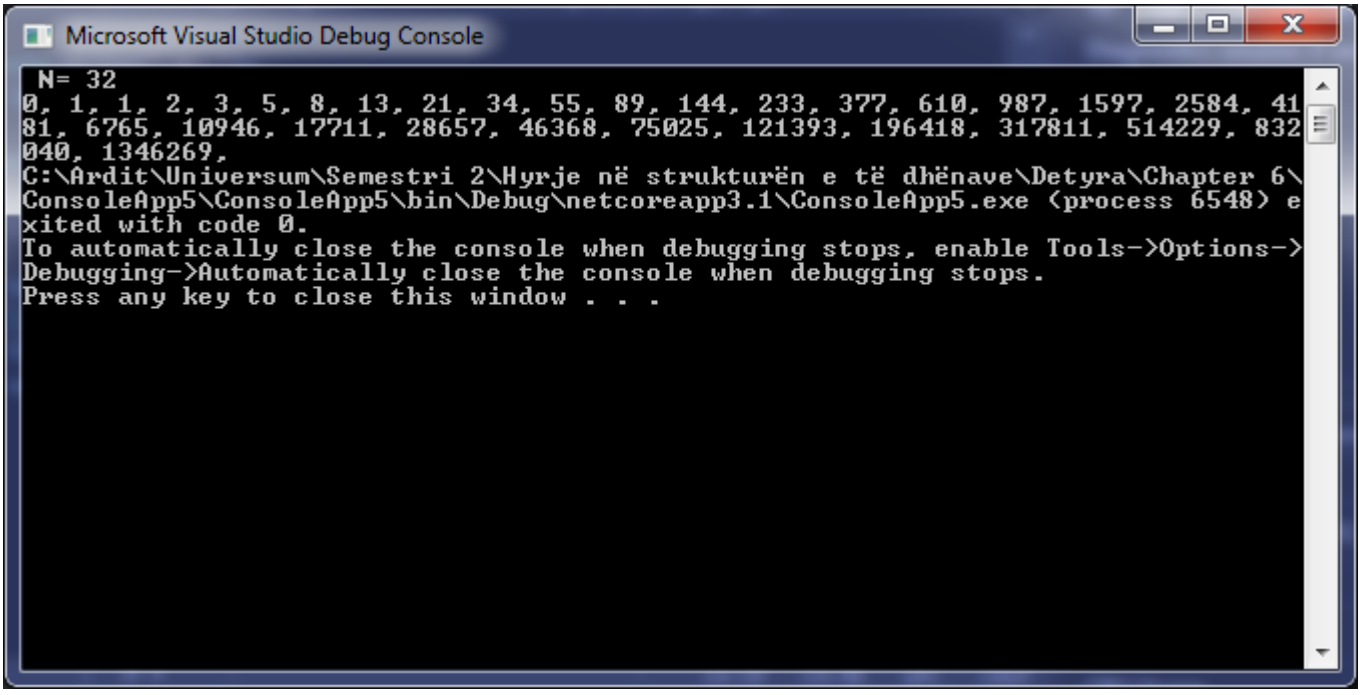
4. Write a program that prints **all possible cards from a standard deck** of cards, without jokers (there are 52 cards: 4 suits of 13 cards).

```
static void Main(string[] args)
        {
            for (int i = 0; i < 4; i++)
            {
                if (i != 0) Console.WriteLine();

                for (int j = 0; j < 13; j++)
                {
                    switch (i)
                    {
                        case 0: Console.Write("Hearts "); break;
                        case 1: Console.Write("Diamonds "); break;
                        case 2: Console.Write("Spades "); break;
                        case 3: Console.Write("Clubs "); break;
                    }
                    switch (j)
                    {
                        case 0: Console.WriteLine("2"); break;
                        case 1: Console.WriteLine("3"); break;
                        case 2: Console.WriteLine("4"); break;
                        case 3: Console.WriteLine("5"); break;
                        case 4: Console.WriteLine("6"); break;
                        case 5: Console.WriteLine("7"); break;
                        case 6: Console.WriteLine("8"); break;
                        case 7: Console.WriteLine("9"); break;
                        case 8: Console.WriteLine("10"); break;
                        case 9: Console.WriteLine("J"); break;
                        case 10: Console.WriteLine("Q"); break;
                        case 11: Console.WriteLine("K"); break;
                        case 12: Console.WriteLine("A"); break;
                    }
                }
            }
            Console.ReadLine();
        }
    }
}
```


C:\Ardit\Universum\Semestri 2\Hyrje në strukturën e të dhënave\Detyra\Chapter 6\ConsoleApp4\...
Hearts 10
Hearts J
Hearts Q
Hearts K
Hearts A
Diamonds 2
Diamonds 3
Diamonds 4
Diamonds 5
Diamonds 6
Diamonds 7
Diamonds 8
Diamonds 9
Diamonds 10
Diamonds J
Diamonds Q
Diamonds K
Diamonds A
Spades 2
Spades 3
Spades 4
Spades 5
Spades 6
Spades 7
Spades 8
Spades 9
Spades 10
Spades J
Spades Q
Spades K
Spades A
Clubs 2
Clubs 3
Clubs 4
Clubs 5
Clubs 6
Clubs 7
Clubs 8
Clubs 9
Clubs 10
Clubs J
Clubs Q
Clubs K
Clubs A

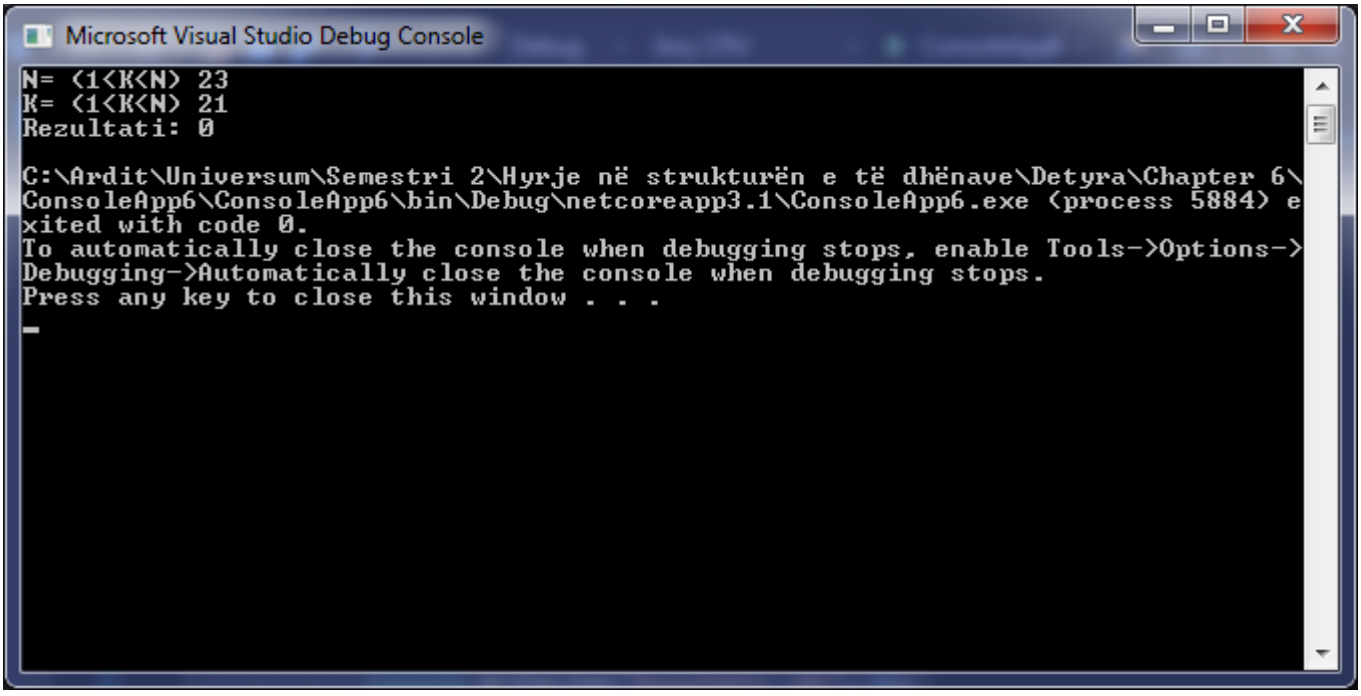
5. Write a program that reads from the console number N and print the sum of the first N members of the **Fibonacci sequence**: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, …

```
using System;

namespace ConsoleApp5
{
    class Program
    {
        static void Main(string[] args)
        {
            int firstN = 0, secondN = 1, thirdN = 0;

            Console.Write(" N= ");
            int length = Int32.Parse(Console.ReadLine());

            Console.Write("0, 1,");

            for (int i = 2; i < length; i++)
            {
                thirdN = firstN + secondN;
                Console.Write(" {0},", thirdN);
                firstN = secondN;
                secondN = thirdN;
            }
        }
    }
}
```



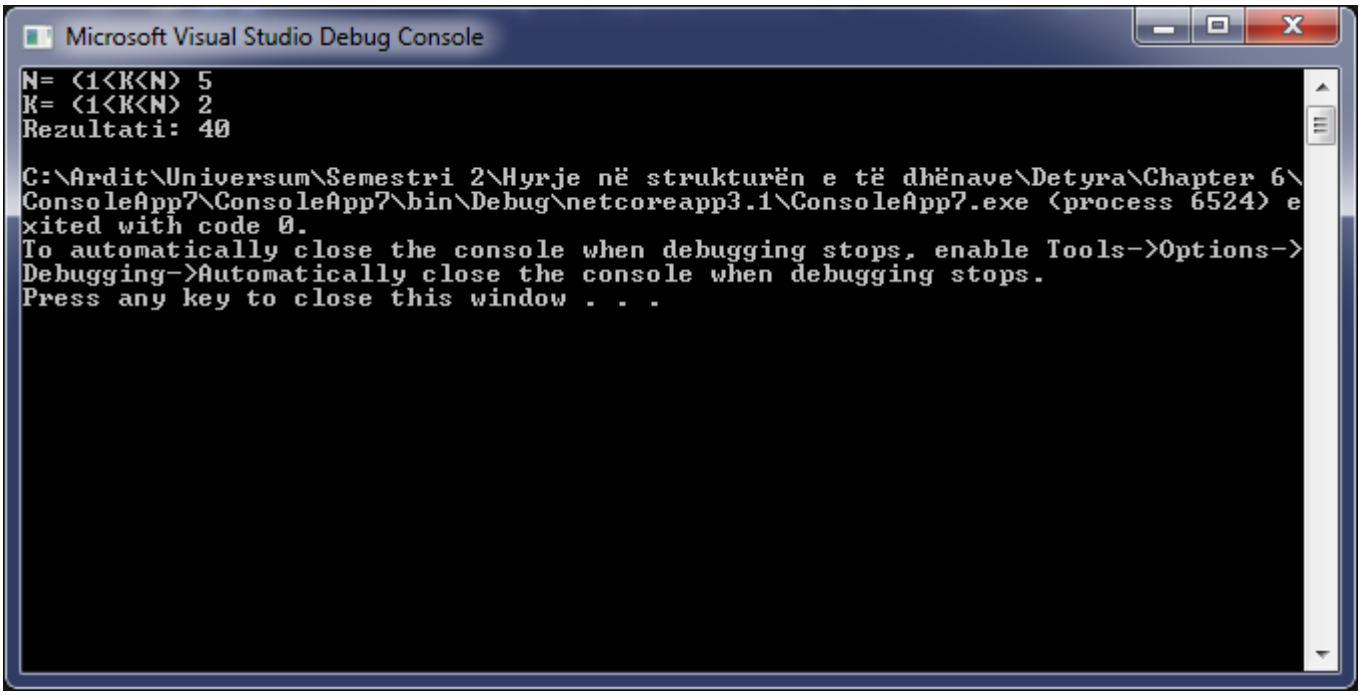
6. Write a program that calculates **N!/K!** for given N and K (1 < K < N).

```
using System;

namespace ConsoleApp6
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.Write("N= (1<K<N) ");
            int n = Int32.Parse(Console.ReadLine());
            Console.Write("K= (1<K<N) ");
            int k = Int32.Parse(Console.ReadLine());

            for (int i = n - 1; i > 0; i--)
            {
                n *= i;
            }

            for (int i = k - 1; i > 0; i--)
            {
                k *= i;
            }

            n /= k;
            Console.WriteLine("Rezultati: {0}", n);
        }
    }
}
```



7. Write a program that calculates **N!*K!/(N-K)!** for given N and K (1 < K < N).

```
using System;

namespace ConsoleApp7
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.Write("N= (1<K<N) ");
            int n = Int32.Parse(Console.ReadLine());
            Console.Write("K= (1<K<N) ");
            int k = Int32.Parse(Console.ReadLine());
            int nMinusK = n - k;

            for (int i = n - 1; i > 0; i--) n *= i;
            for (int i = k - 1; i > 0; i--) k *= i;
            for (int i = nMinusK - 1; i > 0; i--) nMinusK *= i;

            Console.WriteLine("Rezultati: {0}", n * k / nMinusK);
        }
    }
}
```

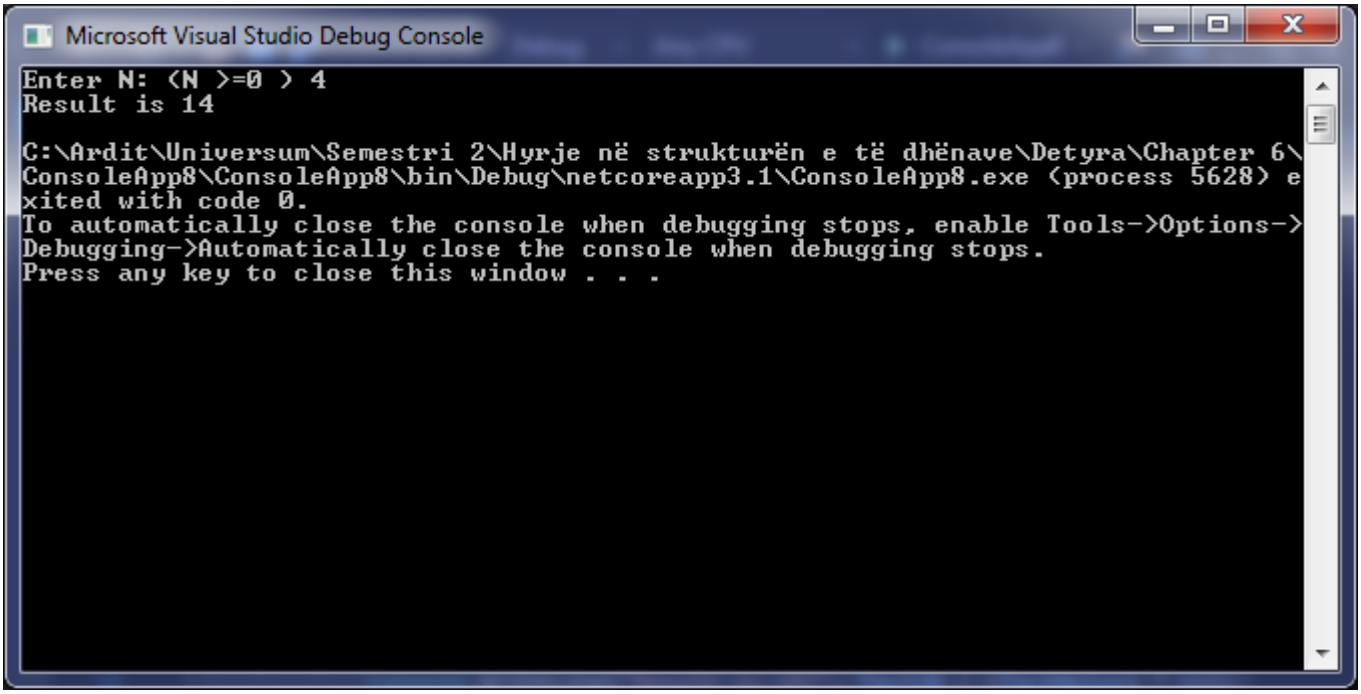


8. In combinatorics, the Catalan numbers are calculated by the following formula:

$$C_n = \frac{1}{n+1}\binom{2n}{n} = \frac{(2n)!}{(n+1)!n!}$$, for n ≥ 0.

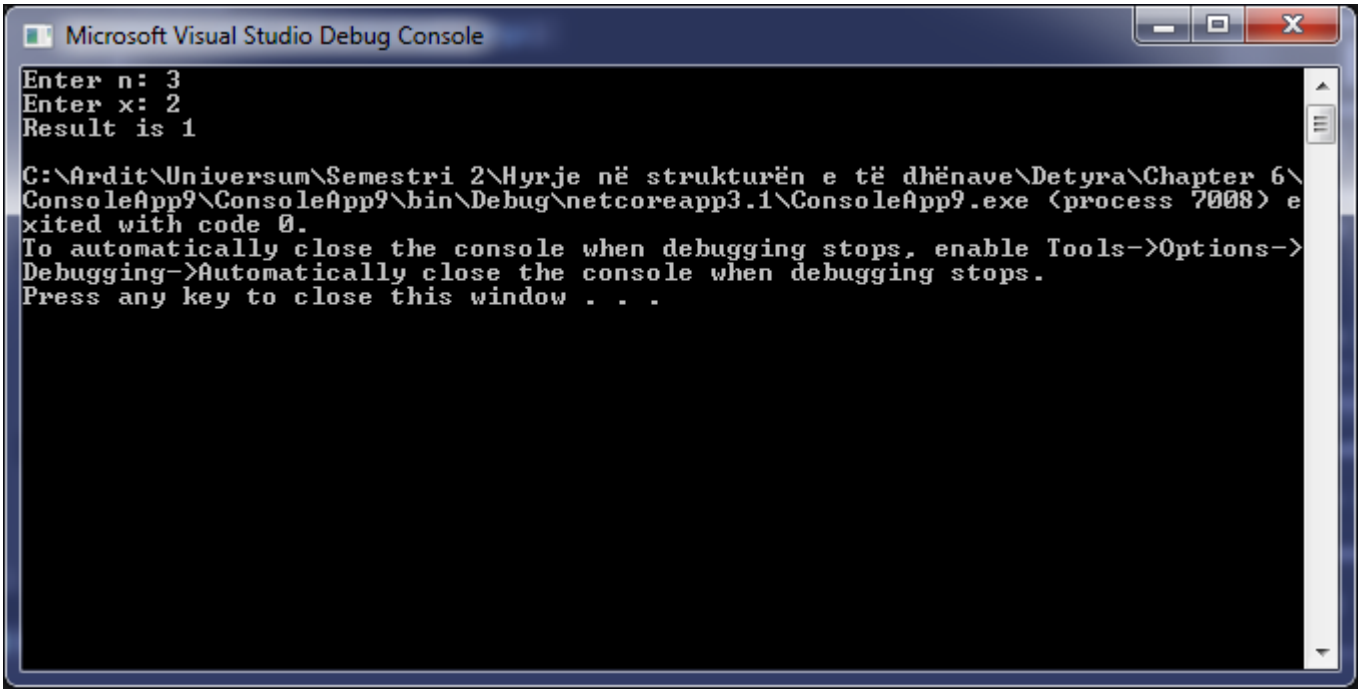
Write a program that calculates the n-th Catalan number by given n.

```
using System;

namespace ConsoleApp8
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.Write("Enter N: (N >=0 ) ");
            int n = Int32.Parse(Console.ReadLine());

            int fact2N = 2 * n, factNplus1 = n + 1;

            for (int i = fact2N - 1; i > 0; i--) fact2N *= i;
            for (int i = factNplus1 - 1; i > 0; i--) factNplus1 *= i;
            for (int i = n - 1; i > 0; i--) n *= i;

            Console.WriteLine("Result is {0}", fact2N / (factNplus1 * n));
        }
    }
}
```



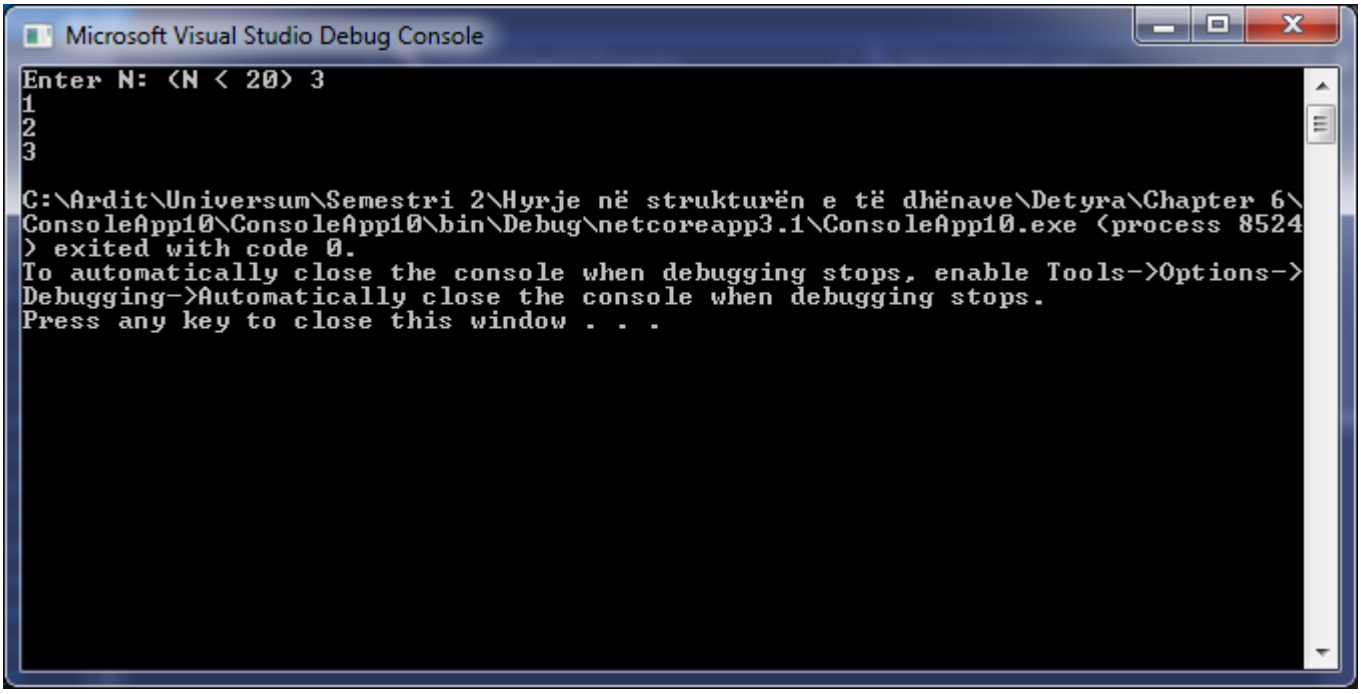
9. Write a program that for a given integers **n** and **x**, **calculates the sum**: $S = 1 + \frac{1!}{x} + \frac{2!}{x^2} + \ldots + \frac{n!}{x^n}$

```
using System;

namespace ConsoleApp9
{
    class Program
    {
        static void Main(string[] args)
        {
            int sum = 1, temp = 1;
            Console.Write("Enter n: ");
            int n = Int32.Parse(Console.ReadLine());
            Console.Write("Enter x: ");
            int x = Int32.Parse(Console.ReadLine());

            for (int i = 1; i <= n; i++)
            {
                temp *= i / x;
                sum += temp;
            }

            Console.WriteLine("Result is {0}", sum);
        }
    }
}
```



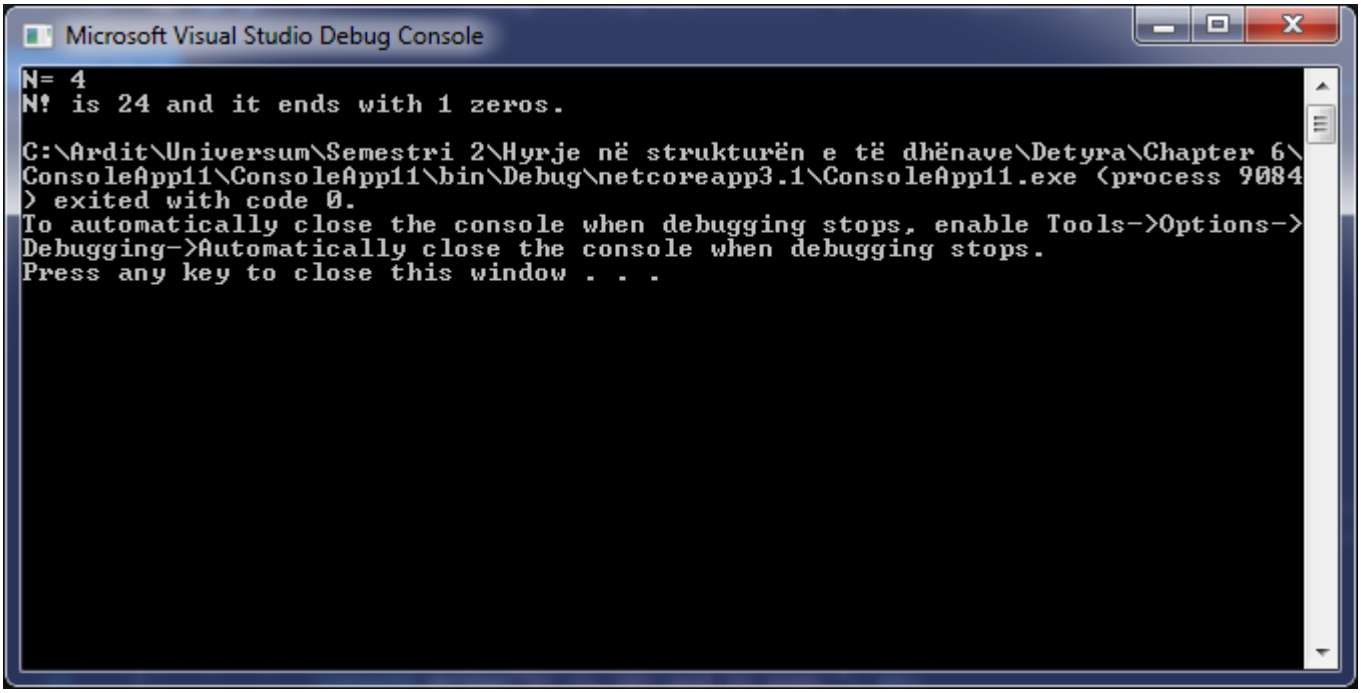
10. Write a program that reads from the console a **positive integer number N** (N < 20) and prints a **matrix** of numbers as on the figures below:

```
using System;

namespace ConsoleApp10
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.Write("Enter N: (N < 20) ");
            int n = Int32.Parse(Console.ReadLine());

            for (int i = 1; i <= n; i++)
            {
                for (int j = i; j <= i; j++)
                {
                    Console.Write("{0} ", j);
                }
                Console.WriteLine();
            }
        }
    }
}
```



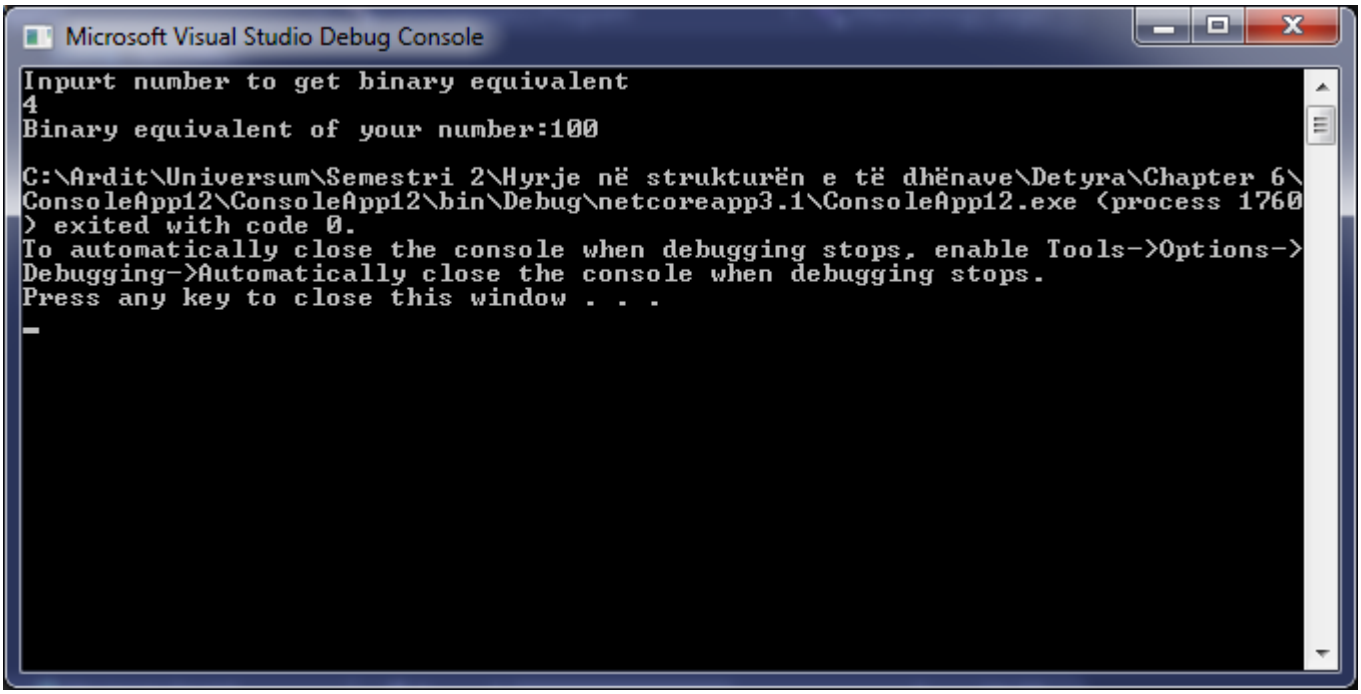
11. Write a program that calculates with **how many zeroes the factorial of a given number ends**. Examples:
N = 10 -> N! = 3628**00** -> 2
N = 20 -> N! = 2432902008176640**000** -> 4

```
using System;

namespace ConsoleApp11
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.Write("N= ");
            decimal n = Int32.Parse(Console.ReadLine());
            int zeroes = 0;

            for (int i = (int)(n - 1); i > 0; i--)
                n *= i;

            Console.Write("N! is {0} and it ends ", n);

            do
            {
                n /= 10;
                zeroes++;
            } while (n % 10 == 0);

            Console.WriteLine("with {0} zeros.", zeroes);
        }
    }
}
```



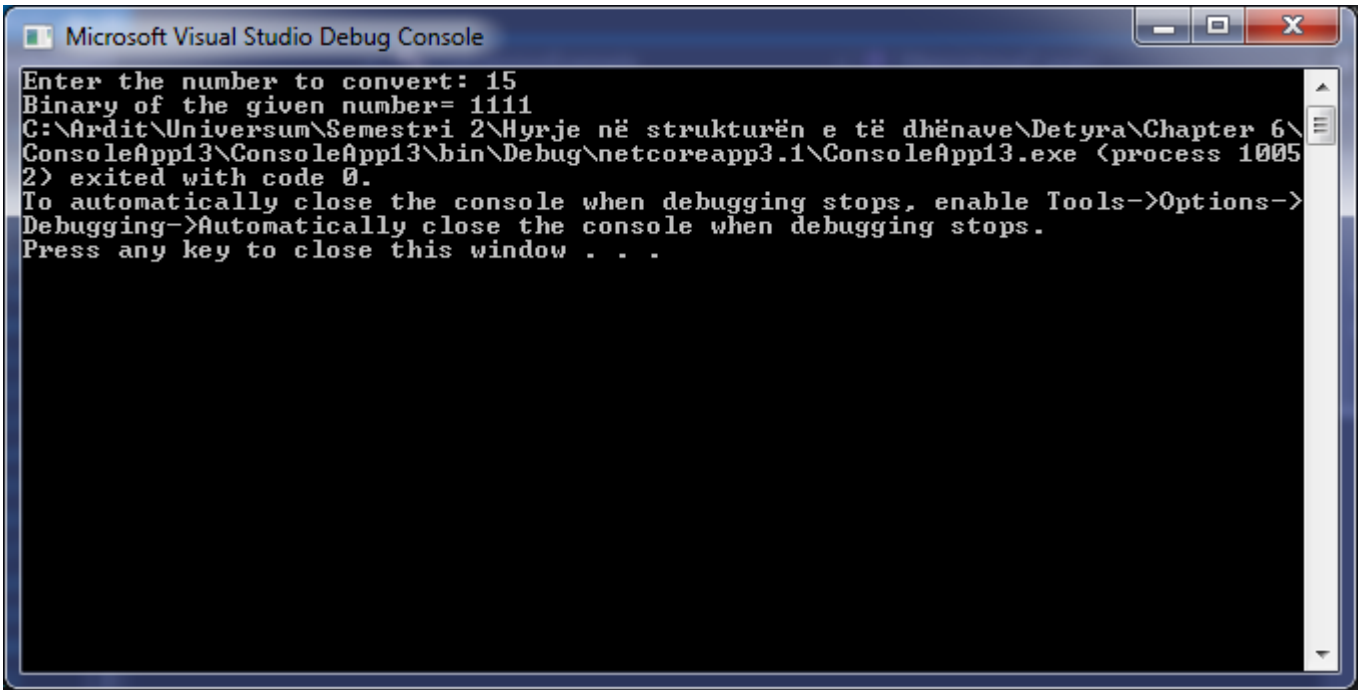
12. Write a program that converts a given number **from decimal to binary notation** (numeral system).

```
using System;

namespace ConsoleApp12
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Inpurt number to get binary equivalent");
            Console.WriteLine("Binary equivalent of your number:" +
Convert.ToString(Convert.ToInt32(Console.ReadLine()), 2));
        }
    }
}
```



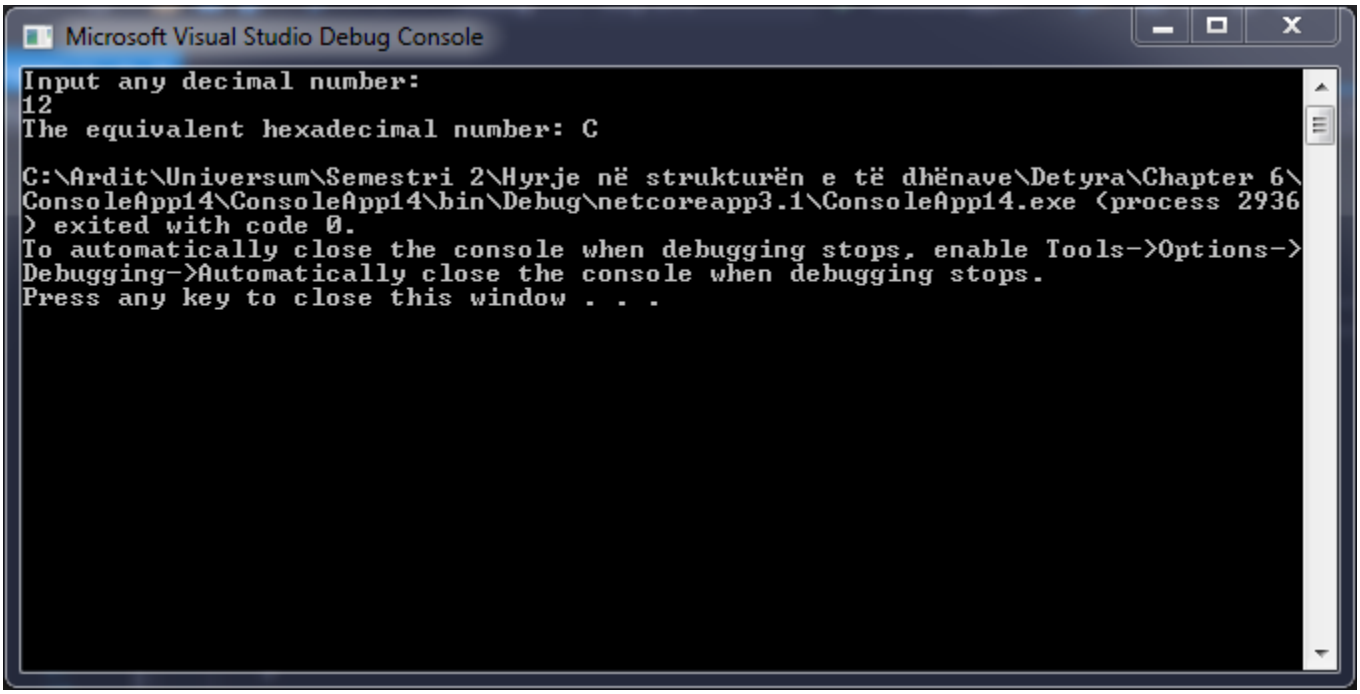
13. Write a program that converts a given number **from binary to decimal notation**.

```
using System;
public class ConversionExample
{
    public static void Main(string[] args)
    {
        int n, i;
        int[] a = new int[10];
        Console.Write("Enter the number to convert: ");
        n = int.Parse(Console.ReadLine());
        for (i = 0; n > 0; i++)
        {
            a[i] = n % 2;
            n = n / 2;
        }
        Console.Write("Binary of the given number= ");
        for (i = i - 1; i >= 0; i--)
        {
            Console.Write(a[i]);
        }
    }
}
```



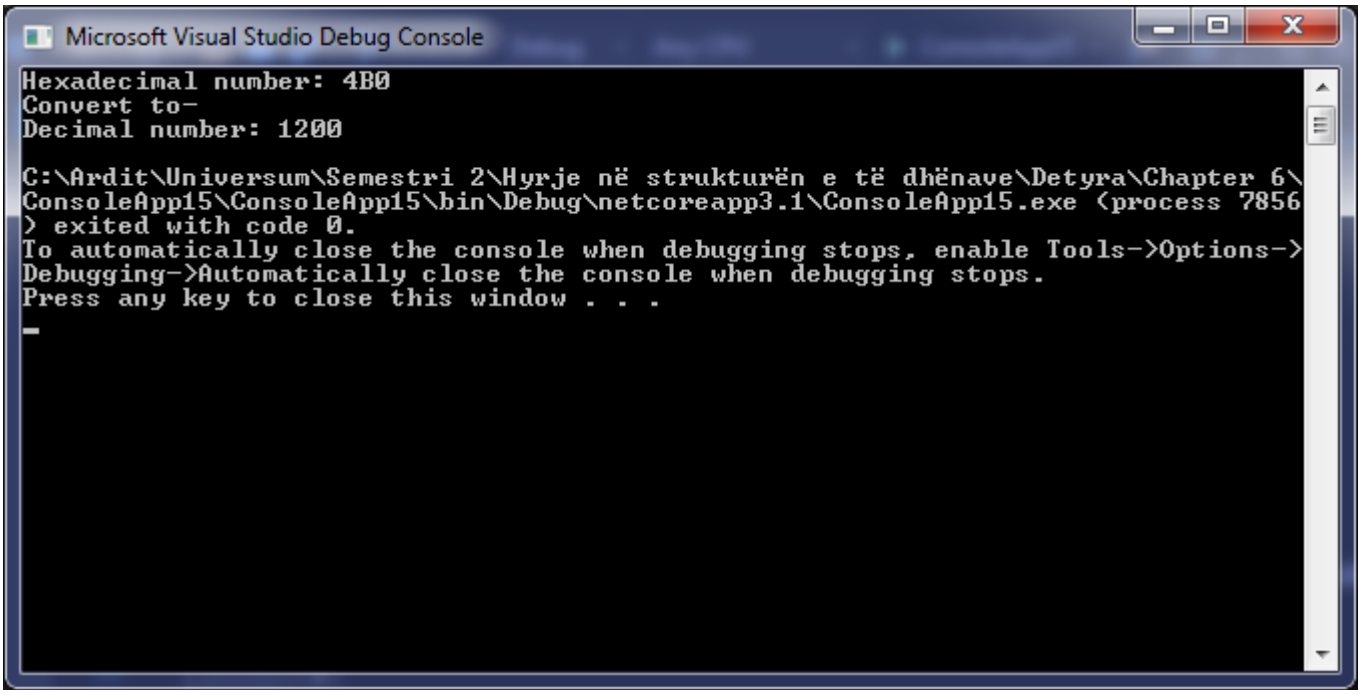
14. Write a program that converts a given number **from decimal to hexadecimal notation**.

```
using System;

namespace ConsoleApp14
{
    using System;
    class program
    {
        public static void Main()
        {
            int x;
            string hexvalue;
            Console.WriteLine("Input any decimal number: ");
            x = Convert.ToInt32(Console.ReadLine());
            hexvalue = x.ToString("X");

            Console.WriteLine("The equivalent hexadecimal number: {0}", hexvalue);
        }

    }
  }
```



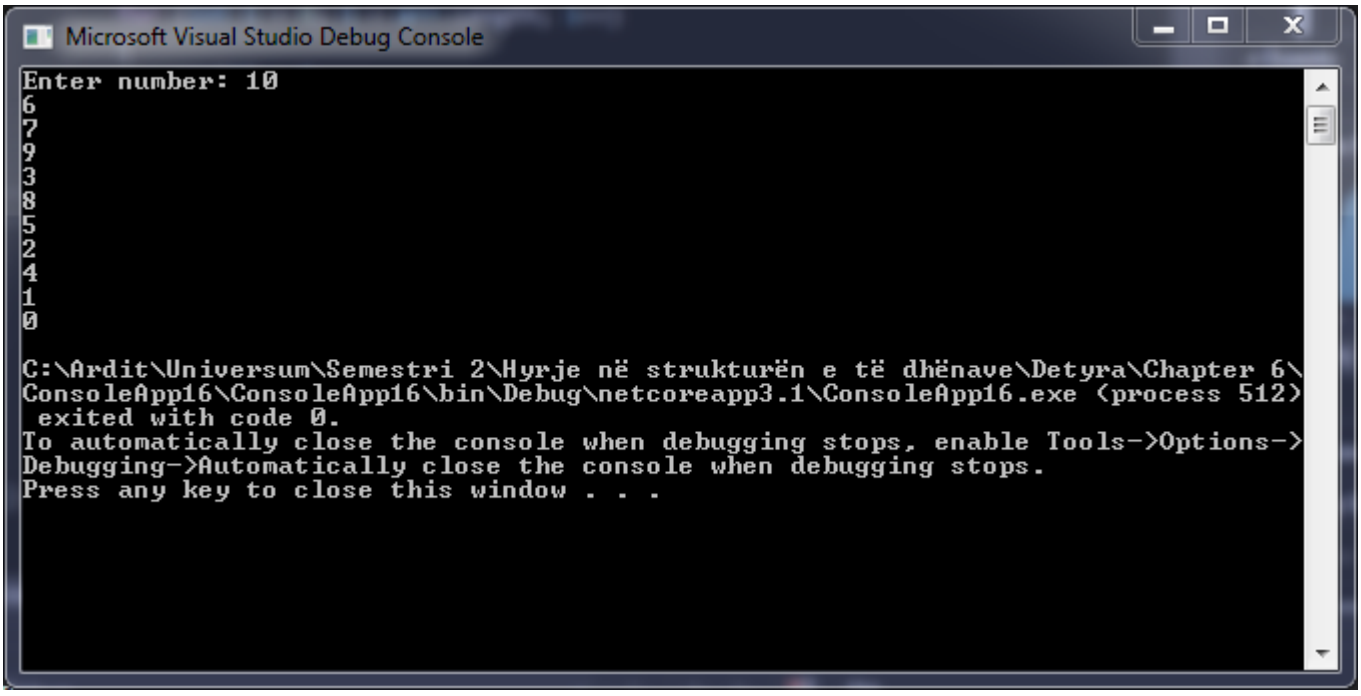
15. Write a program that converts a given number **from hexadecimal to decimal notation**.

```
using System;

namespace ConsoleApp15
{
    class Program
    {
        static void Main(string[] args)
        {
            string hexval = "4B0";
            Console.WriteLine("Hexadecimal number: " + hexval);
            int decValue = int.Parse(hexval, System.Globalization.NumberStyles.HexNumber);
            Console.WriteLine("Convert to-");
            Console.WriteLine("Decimal number: " + decValue);
        }
    }
}
```



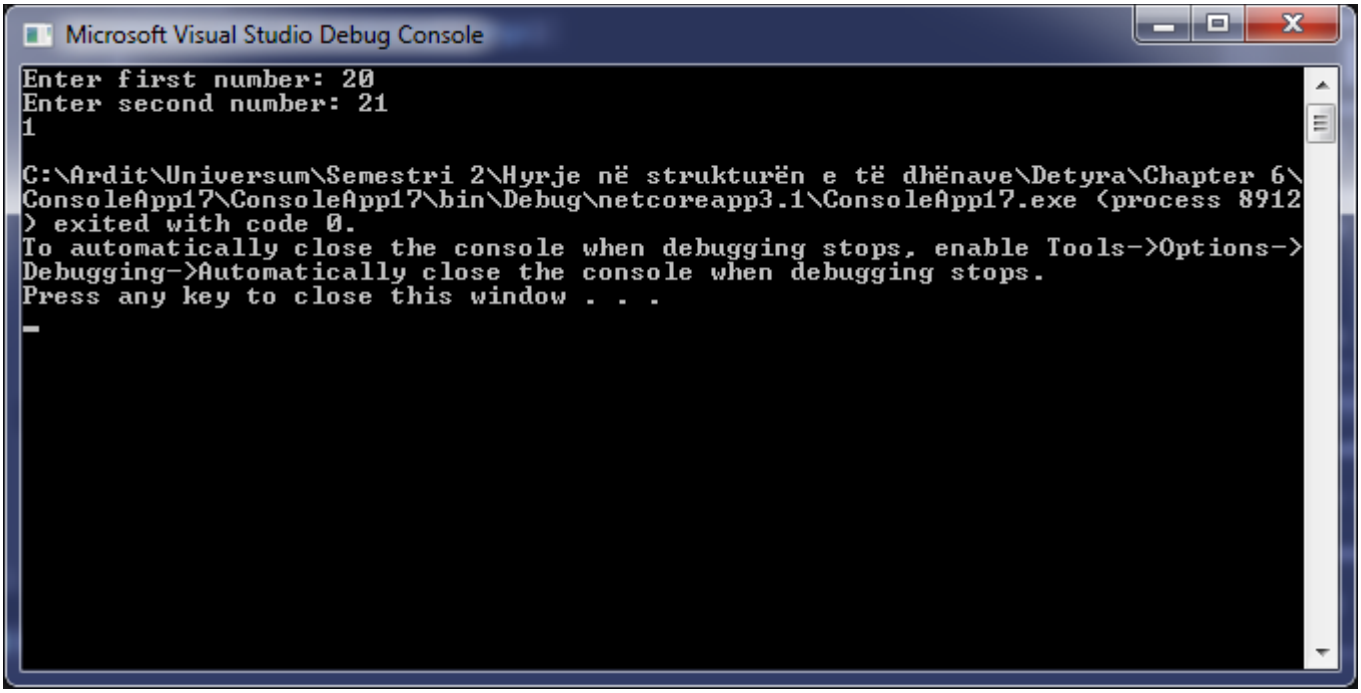
16. Write a program that by a given integer **N** prints the numbers from 1 to N in **random order**.

```
using System;

namespace ConsoleApp16
{
    class Program
    {
        static void Main(string[] args)
        {
            Random rnd = new Random();
            int temp, randomNumber;
            Console.Write("Enter number: ");
            int n = Int32.Parse(Console.ReadLine());
            int[] arr = new int[n];

            for (int i = 0; i < arr.Length; i++)
            {
                arr[i] = i;
            }

            foreach (int i in arr)
            {
                randomNumber = rnd.Next(0, n);
                temp = arr[i];
                arr[i] = arr[randomNumber];
                arr[randomNumber] = temp;
            }

            foreach (int i in arr) Console.WriteLine(arr[i]);
        }
    }
}
```



17. Write a program that given two numbers finds their **greatest common divisor (GCD)**.

```
using System;

namespace ConsoleApp17
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.Write("Enter first number: ");
            int a = Int32.Parse(Console.ReadLine());
            Console.Write("Enter second number: ");
            int b = Int32.Parse(Console.ReadLine());

            while (a != 0 && b != 0)
            {
                if (a > b) a %= b;
                else b %= a;
            }

            if (a == 0) Console.WriteLine(b);
            else Console.WriteLine(a);
        }
    }
}
```



18. Write a program that for a given number n, outputs a matrix in the form of a **spiral**: Example with n=4:

```
using System;

namespace ConsoleApp18
{
    class Program
    {
        static void Main()
        {
            Console.Write("Enter N: ");
            int n = Int32.Parse(Console.ReadLine());
            int[,] matrix = new int[n, n];
            int row = 0, col = 0, direction = 0;

            for (int i = 1; i <= n * n; i++)
            {
                switch (direction)
                {
                    case 0:
                        if (col > n - 1 || matrix[row, col] != 0)
                        {
                            direction = 1;
                            col--;
                            row++;
                        }
                        break;
                    case 1:
                        if (row > n - 1 || matrix[row, col] != 0)
                        {
                            direction = 2;
                            row--;
                            col--;
                        }
                        break;
                    case 2:
                        if (col < 0 || matrix[row, col] != 0)
                        {
                            direction = 3;
                            col++;
                            row--;
                        }
                        break;
                    case 3:
                        if (row < 0 || matrix[row, col] != 0)
                        {
                            direction = 0;
                            row++;
                            col++;
                        }
                        break;
                }

                matrix[row, col] = i;

                switch (direction)
                {
                    case 0: col++; break;
                    case 1: row++; break;
                    case 2: col--; break;
                    case 3: row--; break;
                }
            }
```

```
            for (int i = 0; i < n; i++)
            {
                for (int j = 0; j < n; j++)
                {
                    if (matrix[i, j] < 10) Console.Write("{0}  ", matrix[i, j]);
                    else Console.Write("{0} ", matrix[i, j]);
                }
                Console.WriteLine();
            }
        }
    }
}
```