# Задачи по Информатика зимна ваканция 2019 - Симо Александров 10$^{в}$ клас

## M1-20

Пресмятане на произведението на НОД в двуизмерен масив.

```
using System;

namespace m1_20
{
```

```
    class Program
    {
        static void Main(string[] args)
        {
            // get the arrays' sizes
            int n = int.Parse(Console.ReadLine());
            double mul = 1;

            // declare both array
            double[] arr1 = new double[n];
            double[] arr2 = new double[n];

            // fill the 1st array
            for (int i = 0; i < n; i++)
            {
                arr1[i] =
double.Parse(Console.ReadLine());
            }

            // fill the 2nd array
            for (int i = 0; i < n; i++)
            {
                arr2[i] =
double.Parse(Console.ReadLine());
            }

            // calculate mul of GCD
            for (int i = 0; i < n; i++)
            {
                mul *= GCD(arr1[i], arr2[i]);
            }
```

```
            Console.WriteLine(mul);
        }

        // calculate GCD
        private static double GCD(double a, double
b) => (b == 0) ? a : GCD(b, a % b);
    }
}
```

# Spiral

Преход на двуизмерен масив по спирала, вземането на сумата и най-малката стойност от нея.

```
•using System;

namespace Spiral
{
    class Program
    {
        static void Main(string[] args)
        {
            // size of 2D array
            int n = int.Parse(Console.ReadLine());
            int m = int.Parse(Console.ReadLine());

            // declaration of 2D array
            int[,] spiral = new int[n, m];
```

```
            // fill in the spiral
            for (int i = 0; i < n; i++)
            {
                for (int j = 0; j < m; j++)
                {
                    spiral[i,j] =
int.Parse(Console.ReadLine());
                }
            }


            PrintSpiral(spiral, n, m);
        }

        public static void PrintSpiral(int[,]
spiral, int n, int m)
        {
            // keep result here
            string result = "";

            // sum and min
            int sum = 0;
            int min = int.MaxValue;

            // starting positions
            // given n = 5, m = 5
            //                    v top is 0
            //
            //              1, 2, 3, 5, 9,
            //              6, 9, 7, 1, 3,
```

```
            // left is 0 >> 4, 5, 2, 5, 9, << right
is 4
            //              7, 8, 9, 3, 7,
            //              3, 9, 5, 5, 3
            //
            //                    ^ bottom is 4
            int left = 0;
            int right = n - 1;
            int top = 0;
            int bottom = m - 1;

            // repeat until total area
            while(result.Length < (n * m * 2))
            {

            // add all the values at the current
"top"
            // these >>     1, 2, 3, 5, 9,
            //              6, 9, 7, 1, 3,
            //              4, 5, 2, 5, 9,
            //              7, 8, 9, 3, 7,
            //              3, 9, 5, 5, 3
            //
                for (int i = left; i <= right; i++)
                {
                    int num = spiral[top, i];

                    if(num < min)
                    {
                        min = num;
                    }
```

```
                result += num + " ";
                sum += num;
            }

            // move the top one index "down"
            //             1, 2, 3, 5, 9,
            //  top >>     6, 9, 7, 1, 3,
            //             4, 5, 2, 5, 9,
            //             7, 8, 9, 3, 7,
            //             3, 9, 5, 5, 3
            top++;

            // add all the values at the
current "right"
            //                          V these
            //              1, 2, 3, 5, 9,
            //              6, 9, 7, 1, 3,
            //              4, 5, 2, 5, 9,
            //              7, 8, 9, 3, 7,
            //              3, 9, 5, 5, 3
            //
            for (int i = top; i <= bottom; i++)
            {
                int num = spiral[i, right];

                if(num < min)
                {
                    min = num;
                }

                result += num + " ";
                sum += num;
```

```
                }

                // move the right one index "left"
                //                        V right
                //            1, 2, 3, 5, 9,
                //            6, 9, 7, 1, 3,
                //            4, 5, 2, 5, 9,
                //            7, 8, 9, 3, 7,
                //            3, 9, 5, 5, 3
                right--;

                // add all the values at the
current "bottom"
                //            1, 2, 3, 5, 9,
                //            6, 9, 7, 1, 3,
                //            4, 5, 2, 5, 9,
                //            7, 8, 9, 3, 7,
                //  these >>  3, 9, 5, 5, 3
                //
                for (int i = right; i >= left; i--)
                {
                    int num = spiral[bottom, i];

                    if(num < min)
                    {
                        min = num;
                    }

                    result += num + " ";
                    sum += num;
                }
```

```
                // move the bottom one index "up"
                //            1, 2, 3, 5, 9,
                //            6, 9, 7, 1, 3,
                //            4, 5, 2, 5, 9,
                //  bottom >> 7, 8, 9, 3, 7,
                //            3, 9, 5, 5, 3
                bottom--;

                // prevent repeating
                if (right < left)
                {
                    break;
                }

                // add all the values at the
current "left"
                //      these V
                //            1, 2, 3, 5, 9,
                //            6, 9, 7, 1, 3,
                //            4, 5, 2, 5, 9,
                //            7, 8, 9, 3, 7,
                //            3, 9, 5, 5, 3
                //
                for (int i = bottom; i >= top; i--)
                {
                    int num = spiral[i, left];

                    if(num < min)
                    {
                        min = num;
                    }
```

```
                    result += num + " ";
                    sum += num;
                }
                // move the left one index "right"
                //          left V
                //            1, 2, 3, 5, 9,
                //            6, 9, 7, 1, 3,
                //            4, 5, 2, 5, 9,
                //            7, 8, 9, 3, 7,
                //            3, 9, 5, 5, 3
                left++;
            }

        Console.WriteLine($"Spiral: {result}");
        Console.WriteLine($"Sum: {sum}");
        Console.WriteLine($"Min: {min}");

        }
    }
}
```