

Контест 13

11 янв 2022, 18:48:59

старт: 11 июн 2021, 12:30:00

финиш: 20 июн 2021, 23:59:59

длительность: 9д. 11ч.

начало: 11 июн 2021, 12:30:00

конец: 20 июн 2021, 23:59:59

С. Часть массива

Ограничение времени	2 секунды
Ограничение памяти	64Mb
Ввод	стандартный ввод
Вывод	стандартный вывод

Миша считает, что математика – это искусство. В одной из задач в учебнике по линейной алгебре он увидел такую матрицу, что на пересечении столбца i и строки j стоит произведение $i \cdot j^*$, и посчитал ее очень красивой. Но, к сожалению, матрица в учебнике была очень маленького размера, поэтому чтобы полюбоваться, Миша решил выписать эту матрицу на бумагу и продолжить ее дальше. Как вы, наверное, знаете, целых чисел бесконечно много, поэтому в какой-то момент у Миши закончилась бумага и он устал писать. Чтобы еще хоть немного полюбоваться, Миша просит вас написать программу, которая выводит на экран хотя бы кусочек этой матрицы.

* Фрагмент матрицы из учебника:

```
0 0 0 0
0 1 2 3
0 2 4 6
0 3 6 9
```

Формат ввода

Вводится четыре числа: координаты нижнего (x_1, y_1) и верхнего углов (x_2, y_2) части массива, которую необходимо вывести.

$0 \leq x_1 < x_2 < 2^{31}, 0 \leq y_1 < y_2 < 2^{31}$

Формат вывода

Необходимо вывести часть двумерного массива a , находящуюся между $a[x_1, y_1]$ и $a[x_2 + 1, y_2 + 1]$ в виде матрицы.

Пример

Ввод	Вывод
<div></div>	<div></div>
10	100 110 120 130 140
10	110 121 132 143 154
14	120 132 144 156 168
14	130 143 156 169 182
	140 154 168 182 196

Примечания

В систему необходимо сдавать только файл Program.ArrayPiece.cs.

```
1 using System;
2 using System.Numerics;
3
4 public partial class Program
5 {
6     private static BigInteger[,] GetArrayPart(int x1, int y1, int x2, int y2)
7     {
8         int[] array1 = new int[] { x2 - x1 + 1, y2 - y1 + 1 };
9         int[] array2 = new int[] { x1, y1 };
10
11         BigInteger[,] array = (BigInteger[,])Array.CreateInstance(typeof(BigInteger), array1, array2);
12
13         for (int i = array.GetLowerBound(0); i <= array.GetUpperBound(0); i++)
14             for (int j = array.GetLowerBound(1); j <= array.GetUpperBound(1); j++)
15             {
16                 BigInteger temp = (BigInteger)i * j;
17                 array[i, j] = temp;
18             }
19
20         return array;
21     }
22 }
```