# Lab: Multidimensional Arrays

Problems for exercises and homework for the <https://softuni.bg/courses/java-fundamentals>

You can check your solutions here: <https://judge.softuni.bg/Contests/381/Multidimensional-Arrays-Lab>

## Read, Sort and Print Array

Write a program that:

* Read **array of strings** from the console
* **Sort array** alphabetically
* Print new **array** to the console

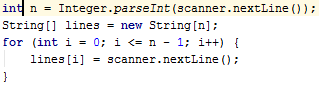
On first line, you will always have number of rows of matrix

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4  Peter  Maria  Katya  Todor | Katya  Maria  Peter  Todor |
| 3  Sofia  London  Washington | London  Sofia  Washington |

### Hints

* Thinks about **Arrays** class
* You might help yourself with the **code** below:



## Sum Matrix Elements

Write program that **read a matrix** from console and print:

* Count of **rows**
* Count of **columns**
* Sum of all **matrix’s elements**

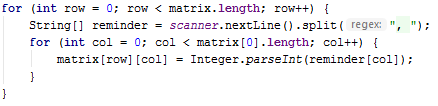
On first line you will get matrix sizes in format **[rows, columns]**

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3, 6  7, 1, 3, 3, 2, , 1 1, 3, 9, 8, 5, 6 4, 6, 7, 9, 1, 0 | 3  6  76 |

### Hints

* One next **[rows]** lines you will get elements for each column separated with coma
* For matrix read see the code below
* Try to use only **foreach()**



## Maximum sum of 2x2 submatrix

Write a program that **read a matrix** from console. Then find biggest sum of **2x2 submatrix** and print it to console.

On first line you will get matrix sizes in format **rows, columns.**

One next **rows** lines you will get elements for each **column** separated with coma.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3, 6  7, 1, 3, 3, 2, 1 1, 3, 9, 8, 5, 6 4, 6, 7, 9, 1, 0 | 9 8  7 9  33 |
| 2, 4  10, 11, 12, 13  14, 15, 16, 17 | 12 13  16 17  58 |

### Hints

* Think about **IndexOutOfBoundsException()**

## Group Numbers

Read a set of numbers and **group** them by their remainder when **dividing to 3** (0, 1 and 2)

One first line, you will get numbers separated with coma

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1, 4, 113, 55, 3, 1, 2, 66, 557, 124, 2 | 3 66  1 4 55 1 124  113 2 557 2 |
| 1, 4, -113, 55, -3, 1, -2, 66, 557, -124, 2 | -3 66  1 4 55 1 -124  -113 -2 557 2 |

### Hints

* Think about how to get **all rows lengths**
* First element in each array will be **easy**, but what will happen with **next numbers we need to add** to each row. Probably you will need one more **array** to save next **index** for each row

## Pascal Triangle

The triangle may be constructed in the following manner: In row 0 (the topmost row), there is a unique nonzero entry 1. Each entry of each subsequent row is constructed by adding the number above and to the left with the number above and to the right, treating blank entries as 0. For example, the initial number in the first (or any other) row is 1 (the sum of 0 and 1), whereas the numbers 1 and 3 in the third row are added to produce the number 4 in the fourth row.

If you want more info about it: <https://en.wikipedia.org/wiki/Pascal's_triangle>

TODO:FIX P5 TESTs

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4 | 1  1 1  1 2 1  1 3 3 1 |
| 15 | 1  1 1  1 2 1  1 3 3 1  1 4 6 4 1  1 5 10 10 5 1  1 6 15 20 15 6 1  1 7 21 35 35 21 7 1  1 8 28 56 70 56 28 8 1  1 9 36 84 126 126 84 36 9 1  1 10 45 120 210 252 210 120 45 10 1  1 11 55 165 330 462 462 330 165 55 11 1  1 12 66 220 495 792 924 792 495 220 66 12 1  1 13 78 286 715 1287 1716 1716 1287 715 286 78 13 1  1 14 91 364 1001 2002 3003 3432 3003 2002 1001 364 91 14 1 |

### Hints

* Input number **n** will be **1 <= n <= 100**
* Think about proper **type** for elements in array
* Don’t be scary to use **more and more Arrays**