Lab: Abstraction

This document defines the exercises for "Java Advanced" course @ Software University. Please submit your solutions (source code) of all below described problems in Judge.

I. Methods and Arrays

1. Calculate Triangle Area Method

Create a method that calculates a triangle area by a given:

- Base
- Height

Return the **area** as an output of the program. Format the result to the **second digit** after the decimal separator.

Examples

Input	Output
4.00 2	Area = 4.00
3 6	Area = 9.00

Hints

- Make you program more readable by using a **Method**

2. Encrypt, Sort and Print Array

Write a program that reads a **sequence of strings** from the console. Encrypt every string by summing:

- The code of **each vowel multiplied by the string length**
- The code of **each consonant divided by the string length**

Sort the **number** sequence alphabetically and print it on the console.

On first line, you will always receive the number of strings you have to read.

Examples

Input	Output	Comments
4 Peter Maria Katya Todor	1032 1071 1168 1532	Peter = 1071 Maria = 1532 Katya = 1032 Todor = 1168
3 Sofia London Washington	1396 1601 3202	Sofia = 1601 London = 1396 Washington = 3202

Hints

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

```
int n = Integer.parseInt(scanner.nextLine());
String[] names = new String[n];
for (int i = 0; i < n; i++) {
    names[i] = scanner.nextLine();
}
```

II. Multidimensional Arrays

3. Sum Matrix Elements

Write a program that **reads a matrix** from the console and prints:

- The count of **rows**
- The count of **columns**
- The sum of all **matrix's elements**

On the first line you will get the dimensions of the matrix in format **{rows, columns}.** On the next lines you will get the elements for each **row** separated with a coma.

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

- Help yourself with the code below for reading the matrix
- Try to use a `foreach`-loop

```
for (int row = 0; row < matrix.length; row++) {
    String[] reminder = scanner.nextLine().split( regex: ", ");
    for (int col = 0; col < matrix[0].length; col++) {
        matrix[row][col] = Integer.parseInt(reminder[col]);
    }
}
```

4. Maximum Sum of 2x2 Submatrix

Write a program that **reads a matrix** from the console. Then find the biggest sum of a **2x2 submatrix.** Print the submatrix and its sum.

On the first line you will get the dimensions of the matrix in format **{rows, columns}.** On the next lines you will get the elements for each **row** separated with a 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

- Ensure that your program doesn't throw an **`IndexOutOfBoundsException()`**

5. Pascals Triangle

Your task is to print the first **N** rows of the Pascal Triangle. You will receive a single integer number **N** as an input.

The Pascal triangle is constructed in the following manner: On 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**.

If you can get more info about it here: https://en.wikipedia.org/wiki/Pascal's_triangle

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

- The Input number **N** will be in range [**1…100**]
- Think about a proper **type** for the elements of the array