

Lab: Arrays, Matrices, Multi-Dimensional Arrays

Problems for in-class lab for the [“JavaScript Fundamentals” course @ SoftUni](#). Submit your solutions in the SoftUni judge system at <https://judge.softuni.bg/Contests/311>.

1. Sum First Last

Write a JS function that calculates and prints the sum of the first and the last elements in an array.

The **input** comes as array of string elements holding numbers.

The **output** is the return value of your function.

Examples

Input	Output
['20', '30', '40']	60

Input	Output
['5', '10']	15

2. Even Position Element

Write a JS function that finds the elements at even positions in an array.

The **input** comes as array of string elements holding numbers.

The **output** is the return value of your function. Collect all elements in a string, separated by space.

Examples

Input	Output
['20', '30', '40']	20 40

Input	Output
['5', '10']	5

3. Negative / Positive Numbers

Write a JS function that processes the elements in an array one by one and produces a new array. Prepend each negative element at the front of the result and append each positive (or 0) element at the end of the result.

The **input** comes as array of string elements holding numbers.

The **output** is printed on the console, each element on a new line.

Examples

Input	Output
['7', '-2', '8', '9']	-2 7 8 9

Input	Output
['3', '-2', '0', '-1']	-1 -2 3 0

4. First and Last K Numbers

Write a JS function that prints the first **k** and the last **k** elements from an array of numbers.

The **input** comes as array of string elements holding numbers. The first element represents the number **k**, all other elements are from the array that needs to be processed.

The **output** is printed on the console on two lines. On the first line print the **first k** elements, separated by space. On the second line print the **last k** elements, separated by space.

Examples

Input	Output	Input	Output
['2', '7', '8', '9']	7 8 8 9	['3', '6', '7', '8', '9']	6 7 8 7 8 9

5. Last K Numbers Sequence

You are given two integers **n** and **k**. Write a JS function that generates and prints the following sequence:

- The first element is 1
- Every following element equals the sum of the previous **k** elements
- The length of the sequence is **n** elements

The **input** comes as array of two string elements holding numbers. The first element represents the number **n**, and the second - the number **k**.

The **output** is printed on the console on a single line, separated by space.

Examples

Input	Output	Input	Output
['6', '3']	1 1 2 4 7 13	['8', '2']	1 1 2 3 5 8 13 21

Explanation

The 2nd element (1) is the sum of the 3 elements before it, but there is only 1, so we take that. The third element, is the sum of the first 2 (1 and 1) and the 4th - the sum of 1, 1 and 2. The 5th element is the sum of the 2nd, 3rd and 4th (1, 2 and 4) and so on.

6. Process Odd Numbers

You are given an array of numbers. Write a JS function that prints the elements at odd positions from the array, doubled and in reverse order.

The **input** comes as array of string elements holding numbers.

The **output** is printed on the console on a single line, separated by space.

Examples

Input	Output	Input	Output
['10', '15', '20', '25']	50 30	['3', '0', '10', '4', '7', '3']	6 8 0

7. Smallest Two Numbers

Write a JS function that prints the two smallest elements from an array of numbers.

The **input** comes as array of string elements holding numbers.

The **output** is printed on the console on a single line, separated by space.

Examples

Input	Output
['30', '15', '50', '5']	5 15

Input	Output
['3', '0', '10', '4', '7', '3']	0 3

8. Biggest Element

Write a JS function that finds the biggest element inside a matrix.

The **input** comes as array of string elements. Each element contains the elements from one row of a matrix, separated by space.

The **output** is the return value of your function. Find the biggest element and return it.

Examples

Input	Output
['20 50 10', '8 33 145']	145

Input	Output
['3 5 7 12', '-1 4 33 2', '8 3 0 4']	33

9. Diagonal Sums

A square matrix of numbers comes as an array of strings, each string holding numbers (space separated). Write a JS function that finds the sum at the main and at the secondary diagonals.

The **input** comes as array of string elements. Each element contains the elements from one row of a matrix, separated by space.

The **output** is printed on the console, on a single line separated by space. First print the sum at the main diagonal, then the sum at the secondary diagonal.

Examples

Input	Output
['20 40', '10 60']	80 50

Input	Output
['3 5 17', '-1 7 14', '1 -8 89']	99 25

10. Equal Neighbors

Write a JS function that finds the number of equal neighbor pairs inside a matrix of variable size and type (numbers or strings).

The **input** comes as array of string elements. Each element contains the elements from one row of a matrix, separated by space.

The **output** is return value of you function. Save the number of equal pairs you find and return it.

Examples

Input	Output
['2 3 4 7 0', '4 0 5 3 4', '2 3 5 4 2']	1

Input	Output
['test yes yo ho', 'well done yo 6', 'not done yet 5']	2

'9 8 7 5 4']			
--------------	--	--	--