Arrays -Exercises

1. Sum First and Last Array Element

Write a function that receives an array of strings and prints the sum of the first and last element in that array.

Examples

Input	Output
['20', '30', '40']	60
['10', '17', '22', '33']	43
['11', '58', '69']	80

2. Day of Week

Write a program which receives a **number** and prints the corresponding name of the day of week. If the number is not a valid day, print '**Invalid day!**'

Examples

Input	Output
3	Wednesday
6	Saturday
11	Invalid day!

3. Reverse an Array of Numbers

Receive a number **n** and an **array** of elements, **create** a **new** array with **n** numbers, **reverse** it and print its elements on a single line, space separated.

Examples

Input	Output
3, [10, 20, 30, 40, 50]	30 20 10
4, [-1, 20, 99, 5]	5 99 20 -1
2, [66, 43, 75, 89, 47]	43 66

4. Reverse an Array of Strings

Receive an **array of strings** (space separated values), reverse it, and print its elements. **Swap** elements. For example, the **first element should be last,** and the **last element should be first** etc.

Examples

Input	Output
['a', 'b', 'c', 'd', 'e']	e d c b a
['abc', 'def', 'hig', 'klm', 'nop']	nop klm hig def abc
['33', '123', '0', 'dd']	dd 0 123 33

Hints

Loop to the **half-length** of the array. Create a **function** to swap **two elements** inside an array.

5. Sum Even Numbers

Receive an **array of strings** parse them to **numbers** and sum only the **even** numbers.

Examples

Input	Output
['1','2','3','4','5','6']	12
['3','5','7','9']	0
['2','4','6','8','10']	30

6. Even and Odd Subtraction

Write a program that calculates the difference between the sum of the even and the sum of the odd numbers in an array.

Examples

Input	Output	Comments
[1,2,3,4,5,6]	3	2 + 4 + 6 = 12 1 + 3 + 5 = 9 12 − 9 = 3
[3,5,7,9]	-24	
[2,4,6,8,10]	30	

7. Equal Arrays

Receive **two string arrays** and print on the console whether they are **identical** or not. Arrays are identical if their elements are **equal**. If the arrays are identical find the **sum** of the first one and print on the console the following message: '**Arrays are identical. Sum: {sum}**', otherwise find the **first index** where the arrays **differ** and print on the console following message:

'**Arrays are not identical. Found difference at {index} index**'.

Examples

Input	Output
['10','20','30'], ['10','20','30']	Arrays are identical. Sum: 60
['1','2','3','4','5'], ['1','2','4','4','5']	Arrays are not identical. Found difference at 2 index

['1'], ['10']	Arrays are not identical. Found difference at 0 index

8. Condense Array to Number

Write a program to receive **an array of numbers** and **condense** them by **summing** adjacent couples of elements until a **single number** is obtained. For example, if we have 3 elements {2, 10, 3}, we sum the first two and the second two elements and obtain {2+10, 10+3} = {12, 13}, then we sum again all adjacent elements and obtain {12+13} = {25}.

Input	Output
[2,10,3]	25
[5,0,4,1,2]	35
[1]	1

Hints

While we have more than one element in the array **nums[]**, repeat the following:

- Allocate a new array **condensed[]** of size **nums.length-1**.
- Sum the numbers from **nums[]** to **condensed[]**:
  - **condensed[i] = nums[i] + nums[i+1]**
- **nums[] = condensed[]**

9. Print every N-th Element from an Array

Write a program that collects every element of an array on a given step.

The **input** comes as **array of strings** and **N - the step**.

The **collections** are every element on the **N-th** step **starting from the first one**, until you reach the end of the array. Then, print elements in a row, separated by single space.

Examples

Input	Output
['5', '15', '31', '14', '20'], 2	5 31 20
['dsa', 'asd', demo, 'test'], 2	dsa demo
['1', '2', '3', '4', '5'], 6	1

10. Add and Remove Elements from Array

Write a program that **adds** and **removes** numbers **to/from** an array. You will receive a command which can either be "**add**" or "**remove**".

The **initial number** is **1**. Each input command should **increase that number**, regardless of what it is.

Upon receiving an "**add**" command you should add the current number to your array. Upon receiving the "**remove**" command you should remove the last entered number, currently existent in the array.

The **input** comes as array of strings. Each element holds a **command**.

Print elements in a row, separated by single space. In case of an empty array, just print "**Empty**".

Examples

Input	Output
['add', 'add', 'add', 'add']	1 2 3 4
['add', 'add', 'remove', 'add', 'add']	1 3 4
['remove', 'remove', 'remove']	Empty

11. Rotate Array

The array should be rotated to the right side, meaning that the last element should become the first, upon rotation.

You receive an array of strings and count of rotation you need to perform.

The **output** is the resulting array after the rotations. The elements should be printed on one line, separated by a **single space**.

Examples

Input	Output
['1', '2', '3', '4'], 2	3 4 1 2
['Banana', 'Orange', 'Coconut', 'Apple'], 15	Orange Coconut Apple Banana

12. Extract a Non-Decreasing Subsequence from an Array

Extracts only those numbers that **form a non-decreasing subsequence**. Start from the **first element** and continue to **the end** of the **given array of numbers**. Any number which is **LESS THAN** the **current biggest one** is **ignored**, alternatively if it's equal or higher than the **current biggest one** you set it as the **current biggest one** and you continue to the next number.

The **input** comes as an array of numbers.

The **output** is the processed array after the filtration, which should be a non-decreasing subsequence. The elements should be printed on one line, separated by a **single space**.

Examples

Input	Output
[1, 3, 8, 4, 10, 12, 3, 2, 24]	1 3 8 10 12 24
[1, 2, 3, 4]	1 2 3 4
[20, 3, 2, 15, 6, 1]	20

13. Negative / Positive Numbers

Write a program 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 an array of number elements.

Input	Output		Input	Output
[7, -2, 8, 9]	-2		[3, -2, 0, -1]	-1
	7			-2
	8			3
	9			0

14. First and Last K Numbers

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

The **input** comes as an array of number elements and **k**.

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		3, [6, 7, 8, 9]	6 7 8
	8 9			7 8 9

15. Last K Numbers Sequence

You are given two integers **n** and **k**. Write a program 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 two number arguments. 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

16. Process Odd Numbers

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

The **input** comes as an array of number elements.

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

17. Smallest Two Numbers

Write a program that prints the **two smallest** elements from an **array of numbers**.

The **input** comes as an array of number elements.

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

Examples

Input	Output	Input	Output
[30, 15, 50, 5]	5 15	[3, 0, 10, 4, 7, 3]	0 3

18. List of Products

You will receive an **array of products**. Print a **numbered array** of all the products **ordered by name**.

Example

Input	Output
["Potatoes", "Tomatoes", "Onions", "Apples"]	1.Apples 2.Onions 3.Potatoes 4.Tomatoes

19. Array Manipulations

Write a program that manipulates an **array of numbers**.

- **Add {number}:** add a number to the end of the array
- **Remove {number}:** remove number from the array
- **RemoveAt {index}:** removes number at a given index
- **Insert {number} {index}:** inserts a number at a given index

Print the **final state** of the array (**separated by spaces**)

The **input** comes as **an array of strings**. First input will be a string containing the **array to manipulate**. Every other **command** you receive will be a string.

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

Example

Input	Output
["4 19 2 53 6 43", "Add 3", "Remove 2", "RemoveAt 1", "Insert 8 3"]	4 53 6 8 43 3

20. Longest sequence of identical elements

Write a program that finds the longest sequence of identical elements in an array of integers. If 2 or more sequences have the same length, print the one that is on the "right".

Input	Output
2 2 2 3 4 4 2 2 2 1	2 2 2
1 1 1 2 3 1 3 3 1 1	1 1 1
1 1 2 3 4 5 6 2 2	1 1
4 4 4 4	4 4 4 4

0 1 1 2 2 3 3 4 4 5 5	1 1