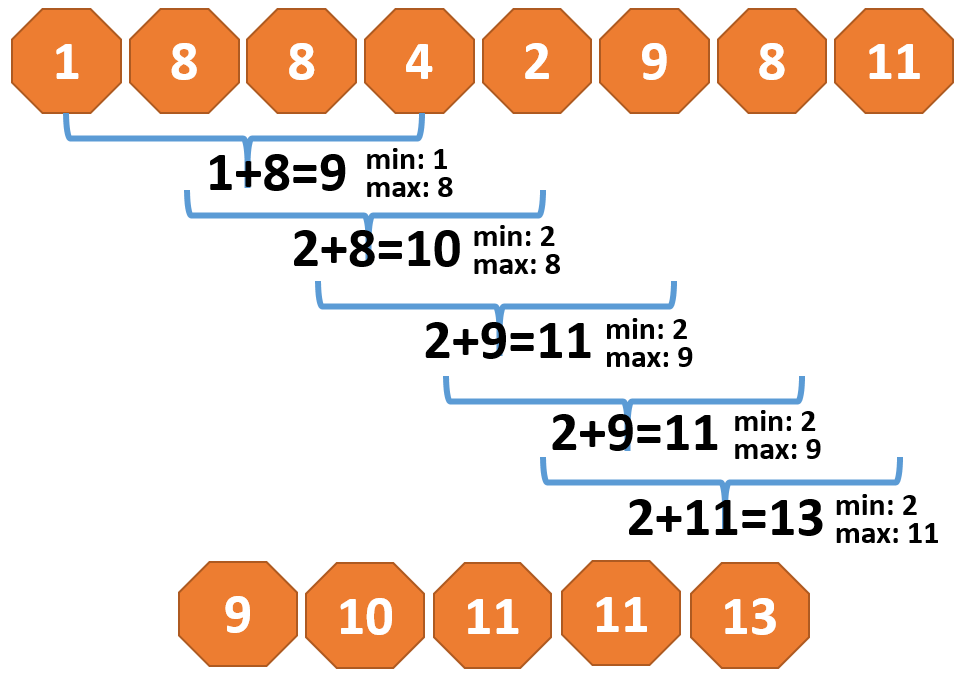
## Problem 1 – MinMax

You are given a list of **N** numbers. You are also given a number **K**.

For each **K** **consecutive numbers** (from left to right) in the given list find **the sum of the minimum and the maximum value** in thesubsequence of these **K** numbers.

Output all the sums obtained **separated by a comma** (‘,’) in order of appearance.

Below is an example where N=8, K=4 and the numbers are 1, 8, 8, 4, 2, 9, 8 and 11. The answer is 9, 10, 11, 10, 13.



### Input

On the first line (params[0]) there will be the number **N.**

On the second line (params[1]) there will be the number **K**.

On the third line (params[2]) there will be a string with **N** integer numbers separated by a single space (‘ ‘).

The input data will always be valid and in the format described. There is no need to check it explicitly.

### Output

Return (or console.log) a single line with a string containing the elements of the obtained list separated by a comma (‘,’).

### Sample solution code (in JavaScript)

**function** *solve*(params) {  
 **var** N = parseInt(params[0]),  
 K = parseInt(params[1]),  
 numbersAsString = params[2];

*// Your solution here*

**console**.log(**'Your answer should be printed on the console'**);  
}

### Constraints

* **N** will be integer number between **1** and **100**, inclusive.
* Each number in the given list will be between -1000000000 and 1000000000.
* **K** will be integer number between **1** and **N**, inclusive.
* Allowed working time for your program: **0.25 seconds**.
* Allowed memory: **32 MB**.

### Examples

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| 4  2  1 3 1 8 | 4,4,9 | 5  3  7 7 8 9 10 | 15,16,18 |  | 8  4  1 8 8 4 2 9 8 11 | 9,10,11,11,13 |