

Hourly Coding Problem

Input file: **standard input**
Output file: **standard output**
Time limit: 6 seconds
Memory limit: 256 megabytes

This problem was asked by Ema.

Given an array of numbers N and an integer k , your task is to split N into k non-empty consecutive partitions such that the maximum sum of any partition is minimized. Output the length of each of the k partitions. If there are multiple solutions, output the solution with the largest lexicographical order.

Solution A is considered to be lexicographically larger than Solution B if there exists an integer $i (1 \leq i \leq n)$, where the first $i - 1$ partitions in A and B have the same length, and the i th partition in A is longer than that in B.

For example, given $N = [5, 1, 0, 2, 7, -3, 4]$ and $k = 3$, you should output $[3, 3, 1]$, since the optimal partition is $[5, 1, 0], [2, 7, -3], [4]$. Note that this example used this format solely for convenience, your program should follow the format described below.

Input

There are multiple test cases. The first line of the input contains an integer T indicating the number of test cases. For each test case:

The first line contains two integers n and k ($1 \leq k \leq n \leq 3 \times 10^5$) indicating the length of the array and the number of partitions.

The next line contains n integer $N = [a_1, a_2, \dots, a_n]$ ($|a_i| \leq 10^9$) indicating the array.

It's guaranteed that the sum of n of all test cases will not exceed 6×10^5 .

Output

For each test case, output one line containing k integers indicating the length of each of the k partitions. Note again that your answer must be the lexicographically largest answer.

Example

standard input	standard output
3	3 3 1
7 3	1 1 2 2
5 1 0 2 7 -3 4	3 3
6 4	
-1 3 -2 4 -3 5	
6 2	
0 -2 0 -1 -1 0	