Problem F: Fantastic compression

Time limit: 3s, memory limit: 512MB.

Franek had one job: to memorize a permutation P of the sequence $(1, 2, \ldots, n)$. This, however, proved too boring. Instead, he compressed the numbers in a new, fantastic way he devised: he took a small integer k and memorized only the sums of all connected k-length fragments of P. In other words, Franek now has a sequence $S = (S_1, S_2, \ldots, S_{n-k+1})$, where:

- $S_1 = P_1 + P_2 + \ldots + P_k$,
- $S_2 = P_2 + P_3 + \ldots + P_{k+1}$,
- $\ldots$
- $S_{n-k+1} = P_{n-k+1} + P_{n-k+2} + \ldots + P_n$.

The method swiftly proved not-so-fantastic, though. First, Franek discovered, to his horror, that sometimes there are several permutations which all compress to the same sequence. Also, he is not sure anymore if he remembered the compressed sequence correctly – the initial permutation may now be lost forever!

Given a compressed sequence S, help Franek find all permutations P which correspond to S.

Input

The first line of input contains the number of test cases z ($1 \leqslant z \leqslant 1000$). The test cases follow, each one in the following format:

The first line of a test case contains the length of the permutation n and the small integer k chosen by Franek ($2 \leqslant n \leqslant 25\,000; 2 \leqslant k \leqslant \min(n, 6)$). The second line contains $n - k + 1$ integers: the elements of the compressed sequence S ($1 \leqslant S_i \leqslant 1\,000\,000$).

The total length of permutations in all testcases does not exceed 250 000.

Output

For every test case, output first the number c of permutations that correspond to the given sequence S. In the next c lines, output these permutations in lexicographic order. Every permutation should be given as n integers in a single line, separated by spaces.

Assume that for the given tests, c is never greater than 1 000.

Example

For the input data	The correct answer is
2 5 3 8 10 12 5 3 10 10 10	2 1 2 5 3 4 2 1 5 4 3 0