DEL13

The bipolar monster strikes again! And the peaceful Line city is his target. As the name suggests, Line city consists of N neighborhoods numbered $1, 2, 3, \dots, N$ arranged from left to right in a line.

The bipolar monster makes bipolar operations on the town. An operation consists of choosing a neighborhood X that is not destroyed yet and that is not the leftmost nor the rightmost with this property, with the purpose of destroying it. However, as the monster is bipolar, right after choosing X, he decides to destroy the 2 yet undestroyed neighborhoods immediate to the left and to the right of X.

In other words, let's assume that at some point, the neighborhoods that are not destroyed yet are, in order from left to right $1 \leq i_1 < i_2 < i_3 < \cdots < i_K \leq N$. X should be one of these neighborhoods, but not the leftmost nor the rightmost. Therefore, $X = i_j$ for j with $1 < j < K$. For this choice of j, the monster will destroy the neighborhoods indexed i_{j-1} and i_{j+1}

Our monster is not that much of a monster as you'd think – the 2 of its personalities don't want to destroy neighborhoods just for fun, they want to make their father proud. The monster knows what exact final configuration of neighborhoods his father would like to see, and so he'd like to make his choices of X in such a way that the final configuration of neighborhoods left undestroyed will please his father. His father's favorite configuration is also a subsequence of the N cities $1 \leq p_1 < p_2 < p_3 < \cdots < p_Q \leq N$.

Sometimes, parents' expectations are a bit unrealistic, and so can be the case with monsters! It is not even guaranteed that with the given operation (which is the only way the monster can destroy cities, given his bipolarity) the desired subsequence of neighborhoods can be achieved. That's why the monster would like to know if he can make a proper choice of his moves so that he will achieve the given subsequence. Not only that, but, as you probably guessed, he wants to know a way of doing so, if possible.

You want to help the monster to make his father proud because he's already been through enough (it's hard to be a bipolar monster). He will give you a list of several cities similar to Line city and of the configurations his father wants to achieve for each such city (both the configuration and the value of N may differ from a city to another). For each city, you need to tell the monster if he can carry out his plan and, if so, how it can do that.

Input

On the first line of the input there will be a number T – the number of cities the monster is considering altering in order to make his father proud.
The next $2T$ lines contain the description of the cities and monster's father's desired final configuration of the cities, in the following format:
$N\ Q$
$p_1, p_2, \dots, p_Q$

Output

For each city you need to print a number R (with $-1 \leq R \leq N$) on a line. R represents the number of operations the monster needs to make in order to achieve the given subsequence of neighborhoods or, if this is impossible to achieve, R should be -1. If it is possible to fulfill the monster's dream, then on the second line, separated by spaces, there should be, in order, the R choices of X that the monster should make. If there are several ways of choosing the values of X, you can print any of them.

The monster is willing to reward you with some points even if you are unable to tell him what moves to make, as long as you tell him whether it's possible or not to fulfill his dream. You should, however, still respect the format. For more details, see the Scoring section.

Scoring

Your output is scored in the following way:
For each test, you will receive a percentage, and this percentage will be used to compute your score as explained in the Subtasks section.

The format should be respected regardless of whether you wanted only to answer the question "is it possible" or not. That is, $-1 \leq R \leq N$ should hold and whenever R is not -1, R should be followed by a separate line with exactly R integer numbers between 1 and N (inclusive). If the format is not respected, even for just one city from an entire test, you will get **0 percent**. Putting $R = 0$ and printing one empty line respects the format and could be used to get credit only for the question "is it possible?" (as described below)

It is considered that your answer to the question "is it possible" is given by whether R is -1 or not.

- If your answer was wrong even for just one city out of an entire test, you will get **0%**
- If your answer was correct at all times (and the moves you gave were not good enough to get full percentage), you will get a percentage of **40%**.

In case that you answered all the "is it possible" questions correctly, you can also receive points for the moves you have provided.

- If, for any city in a test, running the moves you've given, in order, will lead to an invalid move (where you choose an already destroyed city, or you choose a city that was, at the time of your choice, the leftmost or rightmost undestroyed city), **you will not get credit for any of the moves you've printed, and will be scored only for answering correctly the question "is it possible?"**
- If, for one of the cities in a test, after running the moves you've given, the final configuration of neighborhoods doesn't match the desired one, again, **you will not get credit for any of the moves you've printed, and will be scored only for answering correctly the question "is it possible?"**
- If your moves are all valid and lead to the correct final configuration, you will get **100%**

Constraints

- For any city inside a test, $N <= 100000$
- The sum of values N for a test is ≤ 200000

Subtasks

A subtask contains several tests and the score you will get is computed as $\frac{P}{100} \cdot S$ where S is the number of points assigned for the subtask and P is the lowest of the percentages you received for each individual test making up that subtask (percentages are given as explained in the Scoring section)

Subtask	Score	Additional input constraints
1	15	$1 \leq T \leq 512$ and all the cities inside a test have the same value of N with $1 \leq N \leq 9$
2	15	$1 \leq T \leq 2500$ and all the cities inside a test have the same value of N with $1 \leq N \leq 18$
3	15	$1 \leq T \leq 700$ and for all the cities inside a test, the configuration aimed at has at most one segregated neighborhood – there is at most one x so that x is part of the desired final configuration and neither $x - 1$ nor $x + 1$ are part of it
4	25	$1 \leq T \leq 80$ and $1 \leq N \leq 100$
5	30	$1 \leq T \leq 200$

Example

Input	*Output*
4 5 1 3 7 3 1 5 7 7 3 1 2 7 10 4 1 2 5 10	2 3 3 2 3 5 -1 3 8 5 5

In the case of 4th city, the cities left undestroyed, after each move are:

- Initially, 1 2 3 4 5 6 7 8 9 10
- After the first move, 1 2 3 4 5 6 8 10
- After the second move, 1 2 3 5 8 10
- After the third move, 1 2 5 10