

Problem 2 - Seat exchange (seats)

Here at Reply, we encourage knowledge sharing. We also want Replyers to get to know each other, so we organise various social events throughout the year.

One of these events is the Reply Desk Meeting, where Replyers can get to know new people just by sitting at their desk.

At the beginning of the event, N different Replyers are sitting in N different seats, both numbered from 0 to $N - 1$.

After each day, all Replyers move to a new location. In particular, a Replyer sitting in a seat i will move to the seat $P[i]$ the next day.

In this way, as the days go by, each Replyer will always have different neighbours to get to know.

Given the number of days K , write a program to find the final seat configurations.

Input data

The first line of the input file contains an integer T , the number of test cases to solve, followed by T testcases, numbered from 1 to T .

In each test case, the first line contains the two integers N and K : the number of seats and the number of days.

The second line contains N space-separated integers, the array $P[N]$ with the exchanges configuration.

Output data

The output file must contain t lines. For each test case in the input file, the output file must contain a line with the words:

Case #t: $v[0]$ $v[1]$... $v[N - 1]$

where t is the test case number (from 1 to T) followed by N space separated integers: the seats configuration after K days.

Constraints

- $1 \leq T \leq 20$.
- $1 \leq N \leq 100\,000$.
- $1 \leq K \leq 100\,000$.
- At day 0 each Replyer i is seated in his original place i .
- The array $P[N]$ is a permutation of $0, 1, \dots, N - 1$.

Scoring

- **input 1** : $T = 1$, $N \leq 10$ and $K \leq 10$.
- **input 2** : $T = 5$, $N \leq 100$ and $K \leq 100$.
- **input 3** : $T = 10$, $N \leq 1\,000$ and $K \leq 1\,000$.
- **input 4** : $T = 15$, $N \leq 10\,000$ and $K \leq 10\,000$.
- **input 5** : $T = 20$, $N \leq 100\,000$ and $K \leq 100\,000$.

Examples

input	output
1 5 2 0 4 3 1 2	Case #1: 0 2 1 4 3

Explanation

In the first example, after each day we will have the following seats exchanges:

- The Replyer sitting in seat 0 remains there.
- The Replyer sitting in seat 1 moves to seat 4.
- The Replyer sitting in seat 2 moves to seat 3.
- The Replyer sitting in seat 3 moves to seat 1.
- The Replyer sitting in seat 4 moves to seat 2.

The exchanges during the 2 days we be:

- Replyer 0 seated in 0 \rightarrow seated in 0 \rightarrow seated in 0
- Replyer 1 seated in 1 \rightarrow seated in 4 \rightarrow seated in 2
- Replyer 3 seated in 3 \rightarrow seated in 1 \rightarrow seated in 4
- Replyer 4 seated in 4 \rightarrow seated in 2 \rightarrow seated in 3
- Replyer 2 seated in 2 \rightarrow seated in 3 \rightarrow seated in 1

The final configuration of the seats is therefore 0 2 1 4 3.