

## Printer Queue

Source file name: printer.py

Time limit: 2 seconds

The only printer in the computer science students' union is experiencing an extremely heavy workload. Sometimes there are jobs in the printer queue and you may have to wait for hours to print a single page.

Because some jobs are more important than others, the Hacker General has invented and implemented a simple priority system for the print job queue. Now, each job is assigned a priority between 1 and 999 (with 999 being the highest priority, and 1 being the lowest), and the printer operates as follows.

- The first job  $J$  in queue is taken from the queue.
- If there is some job in the queue with a higher priority than job  $J$ , then move  $J$  to the end of the queue without printing it.
- Otherwise, print job  $J$  (and do not put it back in the queue).

In this way, all those important muffin recipes that the Hacker General is printing get printed very quickly. Of course, those annoying term papers that others are printing may have to wait to get printed.

Your problem with the new policy is that it has become quite tricky to determine when your print job will actually be completed. You are to write a program that will be given the current queue (as a list of priorities) as well as the position of your job in the queue, and must then calculate how long it will take until your job is printed, assuming that no additional jobs will be added to the queue. To simplify matters, we assume that printing a job always takes exactly one minute, and that adding and removing jobs from the queue is instantaneous.

### Input

One line with a positive integer: the number of test cases. Then for each test case:

- One line with two integers  $n$  and  $m$ , where  $n$  is the number of jobs in the queue ( $1 \leq n \leq 1000$ ) and  $m$  is the position of your job ( $0 \leq m < n$ ). The first position in the queue is number 0, the second is number 1, ....
- One line with  $n$  integers in the range 1 to 999, giving the priorities of the jobs in the queue. The first integer gives the priority of the first job, the second integer the priority of the second job, ....

*The input must be read from standard input.*

### Output

For each test case, print one line with a single integer; the number of minutes until your job is completely printed, assuming that no additional print jobs will arrive.

*The output must be written to standard output.*

Sample Input	Sample Output
3	1
1 0	2
5	5
4 2	
1 2 3 4	
6 0	
1 1 9 1 1 1	