Problem 7. Scheduler

Input file:	input.txt
Output file:	output.txt
Time limit:	2 seconds
Memory limit:	256 megabytes

There are T processes running in the multitasking operating system ¡¡Squirrel OS¿¿. Each of the T processes has a given priority p_i, which affects how often the process is run. Since the system only has a single core in a single processor to run things, it is facing a challenge of distributing the CPU time between these processes, taking their priorites into account.

The algorithm of defining which process is run at each time moment can be described in the following way. For each process, in addition to the priority p_i, there is also a counter t_i. Initially all t_i equal 0. Then every second:

1. processes with the maximum value of $p_i + t_i$ are chosen.
2. among such processes, the process with the minimum number i is chosen.
3. the chosen process i is run for one second.
4. for the chosen process i the value t_i is set to 0.
5. for all other processes, the value t_i is increased by 1.

Model the work of the operating system for T seconds and calculate for how many seconds each process was run. Assume that all calculations and switches between processes are instant, so the running time for each process in seconds is an integer.

Input

The first line contains two space-separated integers N and T — the number of processes in the operating system ($1 \le N \le 10^5$) and the number of seconds to be modeled ($1 \le T \le 10^6$).

The second line contains N space-separated integers p_i — the process priorities ($0 \le p_i \le 10^5$).

Output

In the only line of the output file, print N space-separated integers — for how many seconds each of the processes was run.

Examples

input.txt	output.txt
3 10 3 4 5	3 3 4