# B. Garbage Collector

You're implementing a garbage collector for the language **Simplang++**. Its memory is a linear array of $n$ slots. Each slot contains either:

- A non-negative integer (**pointer**) indicating another slot's index, or
- A negative integer (**non-pointer data**).

Given the memory array and a set of root pointers, identify all contiguous blocks of unreachable (garbage) slots. Report each garbage block's starting index and size.

## Input

- The first line contains two integers $n$ and $r$, where $n$ $(1 \le n \le 10^5)$ is the size of the memory array, and $r$ $(0 \le r \le n)$ is the number of root pointers.
- The second line contains $n$ integers $a_0, a_1, \ldots, a_{n-1}$, the memory array description:

  - If $a_i \ge 0$, slot $i$ contains a pointer to slot $a_i$.
  - If $a_i < 0$, slot $i$ contains non-pointer data.

- The third line contains $r$ distinct integers $b_1, b_2, \ldots, b_r$ $(0 \le b_i < n)$, the root pointers' addresses.

## Output

- On the first line, output a single integer $m$, the number of garbage memory blocks.
- Then output $m$ lines, each containing two integers $s_i$ and $c_i$, where:

  - $s_i$: the starting index of the $i^{th}$ garbage memory block.
  - $c_i$: the size (number of contiguous slots) of this garbage block.

- Garbage blocks must be reported in **ascending order** of their starting indices.
- If there are no garbage blocks, output a single line containing $0$.

## Example

| input | Скопировать |
|---|---|

```
10 2
1 -3 3 -1 2 7 -2 -1 9 -1
0 4
```

| output | Скопировать |
|---|---|

```
1
5 5
```