HPC³ 2024
Problem J, English
Defence Configuration
Maximum Points: 75

You are the apprentice to a sorcerer who defends your island city from pirates. For this purpose, he has installed N shields numbered from 0 to $N-1$ around the city. However, your master has recently left for an important task and the pirates have seized this opportunity. They have surrounded the city with M $(2 \leq M \leq 100)$ evenly spaced galleons[1] each of which fires a cannon volley at the city every minute. Each of the shields stretch across some number of trajectories between ships and the city given by the array of integers R where R_i $(0 \leq i < N)$ is the number of trajectories the shield can block at any one time. Shields can only be positioned in such a way that they block trajectories fully from their position.

Formally, if you construct a circle with the city at its centre and divide it into M sectors of equal size, each sector represents the angle of attack for a ship. N arcs that cover R_i sectors exist independently within the circle such that the beginning and end of each arc contacts a radius of a sector.

If a ship fires a cannon volley at a trajectory that is blocked by a shield, it will be deflected, and the city will not be hit. Otherwise, the city is hit.

The shields start in some configuration that is not known to you. Every minute, you may move a shield clockwise or counterclockwise such that it now covers 1 trajectory it previously did not and no longer covers 1 trajectory that it previously did. After this, each pirate ship will fire a volley. However, the shields can only be operated from an underground facility. This means that you do not know the origin location of each volley, though you do know how many hit, and you do not know the positions of each shield.

You must protect the city from the invasion by reconfiguring the shields.

The sum of R is greater than or equal to M.

[1] A type of large military ship.

Notes

- This problem is interactive, for every test case your program will be repeatedly giving output and receiving input that is dependent on past outputs.

Subproblem 1

Each minute will begin with the pirates will firing their cannons, then you having the option to move a shield. Your goal is to configure the shields so that all trajectories are covered by at least 1 shield, meaning no cannon volleys hit.

You will be given the number of hits from the first volley, h. Then, each input/output cycle, you will give the number of a shield and the direction to move it in and will receive the number of hits from the next volley, h.

The city cannot survive more than L $(1 \leq L \leq 1.5 \times 10^4)$ hits.

First Input format

The first line of each input contains 3 integers N, W, h, and L.
The second line of each input contains N integers: The content of array R.

```
N  W  h  L
R[0]  R[1]  R[2]  …  R[N-1]
```

Minute Output format

The first and only line of each output contains 1 integer n and 1 binary value d.

```
n  d
```

Where n is the shield to be moved and d is the direction. If d is 0, the shield will be moved counterclockwise, if d is 1, the shield will be moved clockwise.

Minute Input format

The first and only line of each input contains 1 integer h.

```
h
```

If h is -1, you have exceeded the hit limit, if h is 0, you have solved the test case, otherwise, h is the number of trajectories not protected by a shield.

Example Test Cases

Input 1

```
2 8 4 10
4 4
```

Output M1

```
0 1
```

Input M1

```
3
```

Output M2

```
1 0
```

Input M2

```
2
```

Output M3

```
0 1
```

Input M3

```
1
```

Output M4

```
1 0
```

Input M4

```
0
```

Given that there are 8 sections, 4 are hit, and the shields are both of size 4, it can be deduced the shields on top of each other. Moving the shields to surround the city is trivial once the relative position of the shields are known. Note there are multiple ways to move the shields into a solution position.

Subproblem 2

The defence operates the same as the previous problem except that each of the pirate ships have a non-negative integer number of cannon batteries, each of which fires a volley. Each ship has a value g $(0 \leq g \leq 15)$ that represents how many volleys it fires each minute. You do not know any values of g. Shields block all volleys from trajectories they block regardless of g.

You will be given the number of hits from the first volley, h. Then, each input/output cycle, you will give the number of a shield and the direction to move it in and will receive the number of hits from the next volley, h.

The city cannot survive more than L $(1 \leq L \leq 1.8 \times 10^5)$ hits.

First Input format

The first line of each input contains 4 integers N, W, h, and L.
The second line of each input contains N integers: The content of array R.

```
N  W  h  L
R[0]  R[1]  R[2]  …  R[N-1]
```

Minute Output format

The first and only line of each output contains 1 integer n and 1 binary value d.

```
n  d
```

Where n is the shield to be moved and d is the direction. If d is 0, the shield will be moved counterclockwise, if d is 1, the shield will be moved clockwise.

Minute Input format

The first and only line of each input contains 1 integer h.

```
h
```

If h is -1, you have exceeded the hit limit, if h is 0, you have solved the test case, otherwise, h is the sum of g of trajectories not protected by a shield.

Example Test Cases

Input 1

```
3  5  4  25
1  2  2
```

Output M1

```
0  0
```

Input M1

```
3
```

Output M2

```
1  0
```

Input M2

```
3
```

Output M3

```
1  0
```

Input M3

```
3
```

Output M4

```
0  0
```

Input M4

```
0
```

Moving shield 0 reduces the damage taken by 1. Moving shield 1 causes no change, so it can be deduced that the trajectories it moved out of and in to are already covered. Since shield 0 was just moved and shield 2 cannot stretch around the entire island, the relative clockwise distances to each shield from shield 0 are 0, 0, and 2. This means that moving shield 0 once more counterclockwise covers the island.

Subproblem 3

The defence operates the same as the previous problem except now the pirates are using a special type of shield-piercing ammunition with a strange property: If there is more than 1 shield in its path, it ignores all of them. Formally, trajectories are only considered protected if there is exactly 1 shield.

You will be given the number of hits from the first volley, h. Then, each input/output cycle, you will give the number of a shield and the direction to move it in and will receive the number of hits from the next volley, h.

The city cannot survive more than L $(1 \leq L \leq 7.2 \times 10^5)$ hits.

First Input format

The first line of each input contains 4 integers N, W, h, and L.
The second line of each input contains N integers: The content of array R.

```
N  W  h  L
R[0]  R[1]  R[2]  …  R[N-1]
```

Minute Output format

The first and only line of each output contains 1 integer n and 1 binary value d.

```
n  d
```

Where n is the shield to be moved and d is the direction. If d is 0, the shield will be moved counterclockwise, if d is 1, the shield will be moved clockwise.

Minute Input format

The first and only line of each input contains 1 integer h.

```
h
```

If h is -1, you have exceeded the hit limit, if h is 0, you have solved the test case, otherwise, h is the sum of g of trajectories not protected by 1 shield.

Example Test Cases

Input 1

```
2  5  5  25
2  2
```

Output M1

```
1 1
```

Input M1

```
6
```

Output M2

```
0 1
```

Input M2

```
4
```

Output M3

```
0 1
```

Input M3

```
1
```

Output M4

```
0 1
```

Input M4

```
2
```

Output M5

```
1 1
```

Input M5

```
0
```

Given that there are only shields of size 2 and 2 but 5 total trajectories. It can be assumed that one of the trajectories has a g-value of 0 since the problem must be solvable. This can be used to determine the positions of shields.