Problem C. Pecola or Treat

Problem Description

On a serene Halloween night, bathed in the gentle glow of the moonlight, Pecola arrived at the hallo office, quietly prepared to test her newly developed Usada Virus. This virus possessed a unique capability - it could transform infected individuals into adorable rabbit-like creatures. Pecola's intention was to inject some fun and surprise into the upcoming Halloween party.

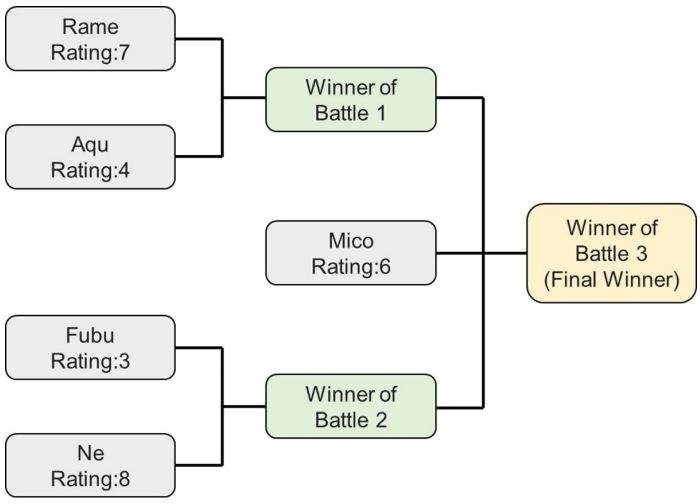
However, as she carefully opened the container containing the virus, a small mishap caused it to slip from her hand, releasing the virus in a mist that swiftly spread throughout the entire office. Before long, every corner of the hallo office was filled with a brief mist, and everyone present had been infected by the magical Usada Virus, transforming into cute rabbit-like creatures.

Pecola found the transformation of everyone quite amusing and let out her signature arrow-like laughter, "HA↑HA↓HA↑HA↓HA↑." However, the abundance of rabbit transformations began to diminish the uniqueness of being a rabbit representative. Consequently, she decided to keep the rabbit form for only one individual and forcibly inject antibodies into the rest. In her role as the internet's giant rabbit, she promptly conceived a new project called "Pecola or Treat" to select members who would retain their rabbit form, serving as compensation for the virus spreading.

The project's specifics were as follows:

- Members would engage in a multiplayer war, and the last person standing would keep their rabbit form.
- Due to the limited size of the arena, **at most** k individuals could participate in each battle.
- In each battle, only one person would emerge as the winner, while the remaining $k-1$ individuals would be forcibly injected with the antibody, losing their rabbit form. The battles would continue repeatedly until only one member retained their rabbit form as the ultimate winner.
- As a participation reward and compensation for being infected by the virus, Pecola would bestow upon each individual $a_i \cdot \mathrm{dis}(i)$ diamonds, where a_i represented each person's rabbitization rating, and $\mathrm{dis}(i)$ indicated the maximum number of battles they needed to participate in.

For instance, when there are 5 members in an arena that can accommodate 3 participants for the battle, here is a possible schedule:



In Rame's case, her Rabbitization rating is 7, and if she advances successfully each time, she would participate in a maximum of 2 battles. Therefore, Pecola would pay her $7 \times 2 = 14$ diamonds.

Following this pattern, Pecola would pay a total of $(7+4+3+8) \times 2 + 6 \times 1 = 50$ diamonds.

With numerous victims, Pecola would have to mine on Neko Neko Island relentlessly to repay them. Although Pecola couldn't change the rabbitization rating of the members, as the organizer, she had the power to determine the schedule. Please assist Pecola in designing a schedule that minimizes the number of diamonds she needs to pay, thereby averting the unfortunate fate of a lifetime of mining on Neko Neko Island.

Input Format

- line 1: n k
- line 2: a_1 a_2 $\ldots$ a_n

Output Format

- line 1: ans : the minimum number of diamonds Pecola needs to pay
- line 2: m : number of battles in your answer
- line $2+i$ ($1 \le i \le m$): k_i $p_{i,1}$ $p_{i,2}$ $\ldots$ p_{i,k_i} : number of players in the i^{th} battle, followed by the list of players
  - Please use $n+i$ to represent the winner of the i^{th} battle.

Your output must satisfy:

- Numbers in $[1, n+m-1]$ must appear exactly once in $p_{i,j}$ for $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, k_i$.
- $2 \le k_i \le n$.
- $1 \le p_{i,j} \le n+i-1$ for $j = 1, 2, \ldots, k_i$.

Constraints

- $1 \le n \le 1\,000\,000$.
- $2 \le k \le 1\,000\,000$.
- $0 \le a_i \le 1\,000\,000\,000$ for $i = 1, 2, \ldots, n$.
- All inputs are integers.

Subtasks

1. (5 points) $n \le k$.
2. (35 points) $k = 2$.
3. (15 points) $n \equiv 1 \bmod (k-1)$.
4. (45 points) No additional constraints.

No.	Testdata Range	Time Limit (ms)	Memory Limit (KiB)
Samples	1-3	2000	262144
1	4-9	2000	262144
2	10-15	2000	262144
3	16-21	2000	262144
4	22-34	2000	262144

Samples

Sample Input 1

```
5 3
4 8 7 6 3
```

This sample input satisfies the constraints of Subtasks 3, 4.

Sample Output 1

```
41
2
3 5 1 4
3 6 3 2
```

Sample Input 2

```
6 3
1 1 1 11 1 1
```

This sample input satisfies the constraints of Subtask 4.

Sample Output 2

```
21
3
3 1 2 3
2 5 6
3 7 4 8
```

Sample Input 3

```
2 2
5 2
```

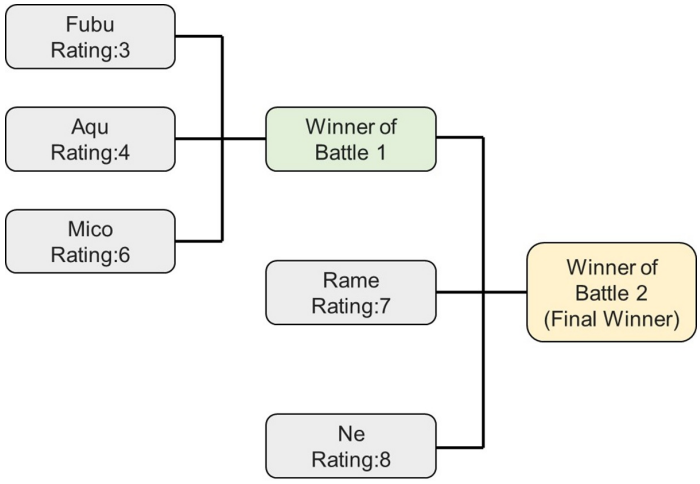
This sample input satisfies the constraints of all the subtasks.

Sample Output 3

```
7
1
2 2 1
```

Notes

One of the best schedule of Sample 1:



One of the best schedule of Sample 2: