C. Beauty Pageant

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

General Payne has a battalion of n soldiers. The soldiers' beauty contest is coming up, it will last for k days. Payne decided that his battalion will participate in the pageant. Now he has choose the participants.

All soldiers in the battalion have different beauty that is represented by a positive integer. The value a_i represents the beauty of the i-th soldier.

On each of k days Generals has to send a detachment of soldiers to the pageant. The beauty of the detachment is the sum of the beauties of the soldiers, who are part of this detachment. Payne wants to surprise the jury of the beauty pageant, so each of k days the beauty of the sent detachment should be unique. In other words, all k beauties of the sent detachments must be distinct numbers.

Help Payne choose k detachments of different beauties for the pageant. Please note that Payne cannot just forget to send soldiers on one day, that is, the detachment of soldiers he sends to the pageant should never be empty.

Input

The first line contains two integers n, k ($1 \le n \le 50$; $1 \le k \le$) — the number of soldiers and the number of days in the pageant, correspondingly. The second line contains space-separated integers $a_1, a_2, ..., a_n$ ($1 \le a_i \le 10^7$) — the beauties of the battalion soldiers.

It is guaranteed that Payne's battalion doesn't have two soldiers with the same beauty.

Output

Print k lines: in the i-th line print the description of the detachment that will participate in the pageant on the i-th day. The description consists of integer c_i ($1 \le c_i \le n$) — the number of soldiers in the detachment on the i-th day of the pageant and c_i distinct integers $p_{1, i}, p_{2, i}, ..., p_{c_i, i}$ — the beauties of the soldiers in the detachment on the i-th day of the pageant. The beauties of the soldiers are allowed to print in any order.

Separate numbers on the lines by spaces. It is guaranteed that there is the solution that meets the problem conditions. If there are multiple solutions, print any of them.

Examples

input
```
3 3
1 2 3
```

output
```
1 1
1 2
2 3 2
```

input
```
2 1
7 12
```

output
```
1 12
```