D. Tanks

Petya sometimes has to water his field. To water the field, Petya needs a tank with exactly V ml of water.

Petya has got N tanks, i-th of them initially containing a_i ml of water. The tanks are really large, any of them can contain any amount of water (no matter how large this amount is).

Also Petya has got a scoop that can contain up to K ml of water (initially the scoop is empty). This scoop can be used to get some water from some tank, and after that pour it all into some tank (it is impossible to get water from multiple tanks without pouring it, or leave some water in the scoop when pouring it). When Petya tries to get some water from a tank, he gets $min(v, K)$ water, where v is the current volume of water in the tank.

Is it possible to obtain a tank with exactly V ml of water using these operations? If it is possible, print a sequence of operations that allows to do it. If there are multiple ways to obtain needed amount of water in some tank, print any of them.

Input

The first line contains 3 integers: N ($2 \le N \le 5000$), K ($1 \le K \le 5000$), and V ($0 \le V \le 10^9$) — the number of tanks, the maximum volume of water the scoop can contain, and the required amount of water in some tank, respectively.

The second line contains N integers a_i ($0 \le a_i \le 10^5$), where a_i is initial volume of water in i-th tank.

Output

If it is impossible to obtain a tank with exactly V ml of water, print `NO`.

Otherwise print `YES` in the first line, and beginning from the second line, print the sequence of operations in the following format:

Each line has to contain 3 numbers denoting a compressed operation: "$cnt\ x\ y$" ($1 \le cnt \le 10^9$, $1 \le x, y \le N$), where x is the index of the tank where we get water, y is the index of the tank where we pour water, and cnt is the number of times we transfer water from tank x to tank y.

The number of these lines **must not exceed $N+5$**.

Examples

input
2 3 5 2 3

output
YES 1 2 1

input
2 3 4 2 3

output
NO

input

```
5 2 0
1 3 5 7 9
```

output

```
YES
2 2 1
3 3 1
4 4 1
5 5 1
```