A. Bmail Computer Network

time limit per test: 4 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Once upon a time there was only one router in the well-known company Bmail. Years went by and over time new routers were purchased. Every time they bought a new router, they connected it to one of the routers bought before it. You are given the values p_i — the index of the router to which the i-th router was connected after being purchased ($p_i < i$).

There are n routers in Boogle in total now. Print the sequence of routers on the path from the first to the n-th router.

Input

The first line contains integer number n ($2 \le n \le 200000$) — the number of the routers. The following line contains $n - 1$ integers $p_2, p_3, \ldots, p_n$ ($1 \le p_i < i$), where p_i is equal to index of the router to which the i-th was connected after purchase.

Output

Print the path from the 1-st to the n-th router. It starts with 1 and ends with n. All the elements in the path should be distinct.

Examples

input
8 1 1 2 2 3 2 5
output
1 2 5 8

input
6 1 2 3 4 5
output
1 2 3 4 5 6

input
7 1 1 2 3 4 3
output
1 3 7

B. DDoS

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

We get more and more news about DDoS-attacks of popular websites.

Arseny is an admin and he thinks that a website is under a DDoS-attack if the total number of requests for a some period of time exceeds $100 \cdot t$, where t — the number of seconds in this time segment.

Arseny knows statistics on the number of requests per second since the server is booted. He knows the sequence $r_1, r_2, \ldots, r_n$, where r_i — the number of requests in the i-th second after boot.

Determine the length of the longest continuous period of time, which Arseny considers to be a DDoS-attack. A seeking time period should not go beyond the boundaries of the segment $[1, n]$.

Input

The first line contains n ($1 \le n \le 5000$) — number of seconds since server has been booted. The second line contains sequence of integers $r_1, r_2, \ldots, r_n$ ($0 \le r_i \le 5000$), r_i — number of requests in the i-th second.

Output

Print the only integer number — the length of the longest time period which is considered to be a DDoS-attack by Arseny. If it doesn't exist print `0`.

Examples

input
5 100 200 1 1 1
output
3

input
5 1 2 3 4 5
output
0

input
2 101 99
output
1

C. Tanya and Colored Candies

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

There are n candy boxes in front of Tania. The boxes are arranged in a row from left to right, numbered from 1 to n. The i-th box contains r_i candies, candies have the color c_i (the color can take one of three values — red, green, or blue). All candies inside a single box have the same color (and it is equal to c_i).

Initially, Tanya is next to the box number s. Tanya can move to the neighbor box (that is, with a number that differs by one) or eat candies in the current box. Tanya eats candies instantly, but the movement takes one second.

If Tanya eats candies from the box, then the box itself remains in place, but there is no more candies in it. In other words, Tanya always eats all the candies from the box and candies in the boxes are not refilled.

It is known that Tanya cannot eat candies of the same color one after another (that is, the colors of candies in two consecutive boxes from which she eats candies are always different). In addition, Tanya's appetite is constantly growing, so in each next box from which she eats candies, there should be strictly more candies than in the previous one.

Note that for the first box from which Tanya will eat candies, there are no restrictions on the color and number of candies.

Tanya wants to eat at least k candies. What is the minimum number of seconds she will need? Remember that she eats candies instantly, and time is spent only on movements.

Input
The first line contains three integers n, s and k ($1 \le n \le 50$, $1 \le s \le n$, $1 \le k \le 2000$) — number of the boxes, initial position of Tanya and lower bound on number of candies to eat. The following line contains n integers r_i ($1 \le r_i \le 50$) — numbers of candies in the boxes. The third line contains sequence of n letters 'R', 'G' and 'B', meaning the colors of candies in the correspondent boxes ('R' for red, 'G' for green, 'B' for blue). Recall that each box contains candies of only one color. The third line contains no spaces.

Output
Print minimal number of seconds to eat at least k candies. If solution doesn't exist, print "-1".

Examples

input
5 3 10 1 2 3 4 5 RGBRR
output
4

input
2 1 15 5 6 RG
output
-1

Note
The sequence of actions of Tanya for the first example:

- move from the box 3 to the box 2;
- eat candies from the box 2;
- move from the box 2 to the box 3;
- eat candy from the box 3;
- move from the box 3 to the box 4;
- move from the box 4 to the box 5;
- eat candies from the box 5.

Since Tanya eats candy instantly, the required time is four seconds.
