Codeforces Round #263 (Div. 2)

A. Appleman and Easy Task

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Toastman came up with a very easy task. He gives it to Appleman, but Appleman doesn't know how to solve it. Can you help him?

Given a $n \times n$ checkerboard. Each cell of the board has either character 'x', or character 'o'. Is it true that each cell of the board has even number of adjacent cells with 'o'? Two cells of the board are adjacent if they share a side.

Input

The first line contains an integer n ($1 \leq n \leq 100$). Then n lines follow containing the description of the checkerboard. Each of them contains n characters (either 'x' or 'o') without spaces.

Output

Print "YES" or "NO" (without the quotes) depending on the answer to the problem.

Examples

input
3 xxo xox oxx
output
YES

input
4 xxxo xoxo oxox xxxx
output
NO

B. Appleman and Card Game

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Appleman has n cards. Each card has an uppercase letter written on it. Toastman must choose k cards from Appleman's cards. Then Appleman should give Toastman some coins depending on the chosen cards. Formally, for each Toastman's card i you should calculate how much Toastman's cards have the letter equal to letter on ith, then sum up all these quantities, such a number of coins Appleman should give to Toastman.

Given the description of Appleman's cards. What is the maximum number of coins Toastman can get?

Input

The first line contains two integers n and k ($1 \leq k \leq n \leq 10^5$). The next line contains n uppercase letters without spaces — the i-th letter describes the i-th card of the Appleman.

Output

Print a single integer – the answer to the problem.

Examples

input
15 10 DZFDFZDFDDDDDDF
output
82

input
6 4 YJSNPI
output
4

Note

In the first test example Toastman can choose nine cards with letter D and one additional card with any letter. For each card with D he will get 9 coins and for the additional card he will get 1 coin.

C. Appleman and Toastman

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

Appleman and Toastman play a game. Initially Appleman gives one group of n numbers to the Toastman, then they start to complete the following tasks:

- Each time Toastman gets a group of numbers, he sums up all the numbers and adds this sum to the score. Then he gives the group to the Appleman.
- Each time Appleman gets a group consisting of a single number, he throws this group out. Each time Appleman gets a group consisting of more than one number, he splits the group into two non-empty groups (he can do it in any way) and gives each of them to Toastman.

After guys complete all the tasks they look at the score value. What is the maximum possible value of score they can get?

Input

The first line contains a single integer n ($1 \le n \le 3 \cdot 10^5$). The second line contains n integers $a_1, a_2, ..., a_n$ ($1 \le a_i \le 10^6$) — the initial group that is given to Toastman.

Output

Print a single integer — the largest possible score.

Examples

input
3 3 1 5
output
26

input
1 10
output
10

Note

Consider the following situation in the first example. Initially Toastman gets group [3, 1, 5] and adds 9 to the score, then he give the group to Appleman. Appleman splits group [3, 1, 5] into two groups: [3, 5] and [1]. Both of them should be given to Toastman. When Toastman receives group [1], he adds 1 to score and gives the group to Appleman (he will throw it out). When Toastman receives group [3, 5], he adds 8 to the score and gives the group to Appleman. Appleman splits [3, 5] in the only possible way: [5] and [3]. Then he gives both groups to Toastman. When Toastman receives [5], he adds 5 to the score and gives the group to Appleman (he will throws it out). When Toastman receives [3], he adds 3 to the score and gives the group to Appleman (he will throws it out). Finally Toastman have added 9 + 1 + 8 + 5 + 3 = 26 to the score. This is the optimal sequence of actions.

D. Appleman and Tree

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

Appleman has a tree with n vertices. Some of the vertices (at least one) are colored black and other vertices are colored white.

Consider a set consisting of k $(0 \le k < n)$ edges of Appleman's tree. If Appleman deletes these edges from the tree, then it will split into $(k + 1)$ parts. Note, that each part will be a tree with colored vertices.

Now Appleman wonders, what is the number of sets splitting the tree in such a way that each resulting part will have exactly one black vertex? Find this number modulo 1000000007 $(10^9 + 7)$.

Input

The first line contains an integer n $(2 \le n \le 10^5)$ — the number of tree vertices.

The second line contains the description of the tree: $n - 1$ integers $p_0, p_1, \ldots, p_{n-2}$ $(0 \le p_i \le i)$. Where p_i means that there is an edge connecting vertex $(i + 1)$ of the tree and vertex p_i. Consider tree vertices are numbered from 0 to $n - 1$.

The third line contains the description of the colors of the vertices: n integers $x_0, x_1, \ldots, x_{n-1}$ (x_i is either 0 or 1). If x_i is equal to 1, vertex i is colored black. Otherwise, vertex i is colored white.

Output

Output a single integer — the number of ways to split the tree modulo 1000000007 $(10^9 + 7)$.

Examples

input
3 0 0 0 1 1
output
2

input
6 0 1 1 0 4 1 1 0 0 1 0
output
1

input
10 0 1 2 1 4 4 4 0 8 0 0 0 1 0 1 1 0 0 1
output
27

E. Appleman and a Sheet of Paper

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

Appleman has a very big sheet of paper. This sheet has a form of rectangle with dimensions $1 \times n$. Your task is help Appleman with folding of such a sheet. Actually, you need to perform q queries. Each query will have one of the following types:

1. Fold the sheet of paper at position p_i. After this query the leftmost part of the paper with dimensions $1 \times p_i$ must be above the rightmost part of the paper with dimensions $1 \times ([\textit{current width of sheet}] - p_i)$.
2. Count what is the total width of the paper pieces, if we will make two described later cuts and consider only the pieces between the cuts. We will make one cut at distance l_i from the left border of the current sheet of paper and the other at distance r_i from the left border of the current sheet of paper.

Please look at the explanation of the first test example for better understanding of the problem.

Input

The first line contains two integers: n and q ($1 \le n \le 10^5$; $1 \le q \le 10^5$) — the width of the paper and the number of queries.

Each of the following q lines contains one of the described queries in the following format:

- "1 p_i" ($1 \le p_i < [\textit{current width of sheet}]$) — the first type query.
- "2 l_i r_i" ($0 \le l_i < r_i \le [\textit{current width of sheet}]$) — the second type query.

Output

For each query of the second type, output the answer.

Examples

input
7 4 1 3 1 2 2 0 1 2 1 2

output
4 3

input
10 9 2 2 9 1 1 2 0 1 1 8 2 0 8 1 2 2 1 3 1 4 2 2 4

output
7 2 10 4 5

Note

The pictures below show the shapes of the paper during the queries of the first example:

After the first fold operation the sheet has width equal to 4, after the second one the width of the sheet equals to 2.