A - Spoon Taking Problem

Time Limit: 2 sec / Memory Limit: 1024 MiB

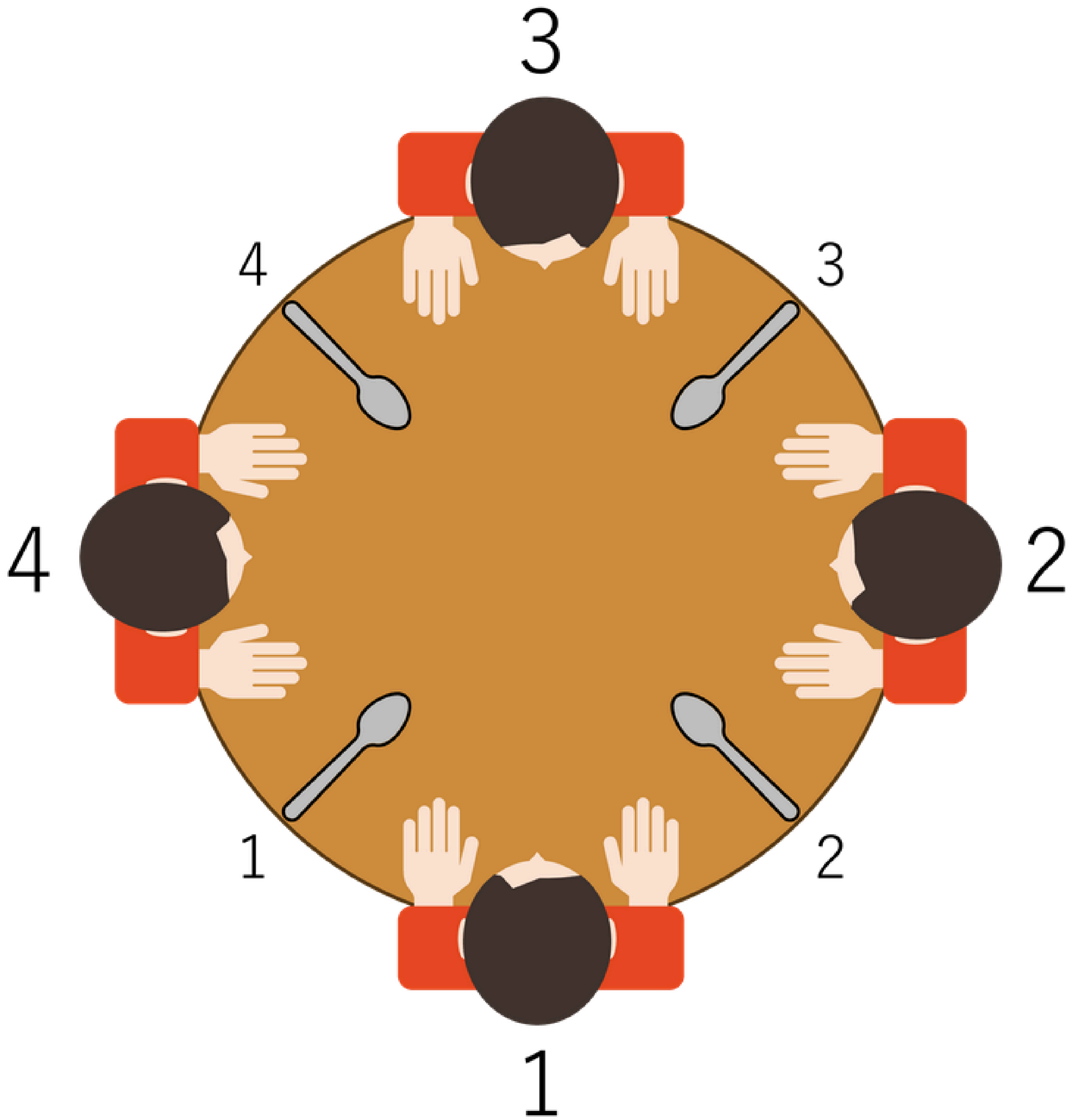
Score: 400 points

Problem Statement

N people are sitting around a round table, and numbered 1 to N in a counterclockwise order. Each person has a dominant hand: left or right.

There are N spoons numbered 1 to N on the round table, with one spoon placed between each pair of adjacent people. For each $1 \leq i \leq N$, to the left and right of person i, there are spoons i and $(i+1)$, respectively. Here, spoon $(N+1)$ refers to spoon 1.

Below is a diagram for $N = 4$.



You are given a permutation $(P_1, \ldots, P_N)$ of $(1, \ldots, N)$. In the order $i = 1, \ldots, N$, person P_i will act as follows:

- If there is a spoon remaining on left or right side, they will take one of them.
  - If there are spoons remaining on both sides, they will take the spoon on the side of their dominant hand.
- Otherwise, they do nothing.

You are also given a string S of length N consisting of `L`, `R`, and `?`. Among the 2^N possible combinations of dominant hands, find how many satisfy all of the following conditions, modulo 998244353:

- If the i-th character of S is `L`, person i is left-handed.
- If the i-th character of S is `R`, person i is right-handed.
- When everyone has finished acting, everyone has taken a spoon.

Constraints

- All input values are integers.
- $2 \leq N \leq 2 \times 10^5$
- $(P_1, \ldots, P_N)$ is a permutation of $(1, \ldots, N)$.
- S is a string of length N consisting of `L`, `R`, and `?`.

Input

The input is given from Standard Input in the following format:

$$N$$
$$P_1 \ \ldots \ P_N$$
$$S$$

Output

Print the answer in a single line.

Sample Input 1

```
3
1 2 3
L??
```

Sample Output 1

```
2
```

When persons 1, 2, and 3 are left-handed, left-handed, and right-handed, respectively, the actions are performed as follows:

- Person 1 starts acting. There are spoons on both sides, so person 1 takes spoon 1 on the left side, which is the same as their dominant hand.
- Person 2 starts acting. There are spoons on both sides, so person 2 takes spoon 2 on the left side, which is the same as their dominant hand.
- Person 3 starts acting. There is no spoon on the right side, but spoon 3 is remaining on the left side, so they take spoon 3. Everyone has finished acting and taken a spoon.

This combination of dominant hands satisfies the conditions. Besides, the conditions are also satisfied when persons $1, 2, 3$ are all left-handed.

Sample Input 2

```
3
1 3 2
R?L
```

Sample Output 2

```
0
```

No combinations of dominant hands satisfy the conditions.

Sample Input 3

```
12
6 2 9 3 1 4 11 5 12 10 7 8
????????????
```

Sample Output 3

```
160
```

B - Parenthesis Arrangement

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score: 400 points

Problem Statement

You are given a string S of length $2N$ consisting of the characters (and). Let S_i denote the i-th character from the left of S.

You can perform the following two types of operations zero or more times in any order:

- Choose a pair of integers (i, j) such that $1 \leq i < j \leq 2N$. Swap S_i and S_j. The cost of this operation is A.
- Choose an integer i such that $1 \leq i \leq 2N$. Replace S_i with (or). The cost of this operation is B.

Your goal is to make S a correct parenthesis sequence. Find the minimum total cost required to achieve this goal. It can be proved that the goal can always be achieved with a finite number of operations.

▶ What is a correct parenthesis sequence?

Constraints

- All input values are integers.
- $1 \leq N \leq 5 \times 10^5$
- $1 \leq A, B \leq 10^9$
- S is a string of length $2N$ consisting of the characters (and).

Input

The Input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \quad A \quad B \\
S
\end{array}
$$

Output

Print the answer in a single line.

Sample Input 1

```
3 3 2
)))(()
```

Sample Output 1

```
5
```

Here is one way to operate:

- Swap S_3 and S_4. S becomes `))()()`. The cost is 3.
- Replace S_1 with `(`. S becomes `()()()`, which is a correct parentheses sequence. The cost is 2.

In this case, we have made S a correct bracket sequence for a total cost of 5. There is no way to make S a correct bracket sequence for less than 5.

Sample Input 2

```
1 175 1000000000
()
```

Sample Output 2

```
0
```

The given S is already a correct bracket sequence, so no operation is needed.

Sample Input 3

```
7 2622 26092458
))()((((()()((
```

Sample Output 3

```
52187538
```

C - Jumping Through Intervals

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score: 600 points

Problem Statement

You are given N pairs of integers $(L_1, R_1), (L_2, R_2), \ldots, (L_N, R_N)$. Here, $L_i \leq R_i$ for all $1 \leq i \leq N$.

A sequence of N integers $A = (A_1, A_2, \ldots, A_N)$ is called a **good integer sequence** if it satisfies the following condition:

- $L_i \leq A_i \leq R_i$ for all $1 \leq i \leq N$.

Find the lexicographically smallest **good integer sequence** A that minimizes $\sum_{i=1}^{N-1} |A_{i+1} - A_i|$.

▶ What is lexicographical order for sequences?

Constraints

- All input values are integers.
- $2 \leq N \leq 5 \times 10^5$
- $0 \leq L_i \leq R_i \leq 10^9$

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \\
L_1 \quad R_1 \\
L_2 \quad R_2 \\
\vdots \\
L_N \quad R_N
\end{array}
$$

Output

Print the answer in a single line in the following format:

$$A_1 \quad A_2 \quad \ldots \quad A_N$$

Sample Input 1

```
4
1 10
8 13
3 4
5 20
```

Sample Output 1

```
8 8 4 5
```

$(A_1, A_2, A_3, A_4) = (8, 8, 4, 5)$ is a good integer sequence. In this case, $\sum_{i=1}^{N-1} |A_{i+1} - A_i| = |8 - 8| + |4 - 8| + |5 - 4| = 5$, which is the minimum value of $\sum_{i=1}^{N-1} |A_{i+1} - A_i|$.

Sample Input 2

```
3
20 24
3 24
1 75
```

Sample Output 2

```
20 20 20
```

Note that when multiple good integer sequences A minimize $\sum_{i=1}^{N-1} |A_{i+1} - A_i|$, you should print the lexicographically smallest of them.

Sample Input 3

```
15
335279264 849598327
446755913 822889311
526239859 548830120
181424399 715477619
342858071 625711486
448565595 480845266
467825612 647639160
160714711 449656269
336869678 545923679
61020590 573085537
626006012 816372580
135599877 389312924
511429216 547865075
561330066 605997004
539239436 921749002
```

Sample Output 3

```
526239859 526239859 526239859 467825612 467825612 467825612 467825612 449656269 449656269 449656269 62
6006012 389312924 511429216 561330066 561330066
```

D - LIS on Tree 2

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score: 700 points

Problem Statement

There is a tree with N vertices numbered 1 to N. The i-th edge of the tree connects vertices u_i and v_i bidirectionally.

For a permutation $P = (P_1, \ldots, P_N)$ of $(1, \ldots, N)$, we define $f(P)$ as follows:

- For each vertex i $(1 \leq i \leq N)$, let $(v_1 = 1, v_2, \ldots, v_k = i)$ be the simple path from vertex 1 to vertex i, and let L_i be the length of a longest increasing subsequence of $(P_{v_1}, P_{v_2}, \ldots, P_{v_k})$. We define $f(P) = \sum_{i=1}^{N} L_i$.

You are given an integer K. Determine whether there is a permutation P of $(1, \ldots, N)$ such that $f(P) = K$. If it exists, present one such permutation.

▶ What is a longest increasing subsequence?

▶ What is a simple path?

Constraints

- All input values are integers.
- $2 \leq N \leq 2 \times 10^5$
- $1 \leq K \leq 10^{11}$
- $1 \leq u_i, v_i \leq N$
- The given graph is a tree.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \quad K \\
u_1 \quad v_1 \\
\vdots \\
u_{N-1} \quad v_{N-1}
\end{array}
$$

Output

If there is no permutation P such that $f(P) = K$, print `No`.

If there is a permutation P such that $f(P) = K$, print it in the following format:

```
Yes
P_1 ... P_N
```

If multiple permutations P satisfy the condition, any of them will be accepted.

Sample Input 1

```
5 8
1 2
2 3
2 4
4 5
```

Sample Output 1

```
Yes
3 2 1 4 5
```

If $P = (3, 2, 1, 4, 5)$, then $f(P)$ is determined as follows:

- The simple path from vertex 1 to vertex 1 is (1), and the length of the longest increasing subsequence of $(P_1) = (3)$ is 1. Thus, $L_1 = 1$.
- The simple path from vertex 1 to vertex 2 is $(1, 2)$, and the length of the longest increasing subsequence of $(P_1, P_2) = (3, 2)$ is 1. Thus, $L_2 = 1$.
- The simple path from vertex 1 to vertex 3 is $(1, 2, 3)$, and the length of the longest increasing subsequence of $(P_1, P_2, P_3) = (3, 2, 1)$ is 1. Thus, $L_3 = 1$.
- The simple path from vertex 1 to vertex 4 is $(1, 2, 4)$, and the length of the longest increasing subsequence of $(P_1, P_2, P_4) = (3, 2, 4)$ is 2. Thus, $L_4 = 2$.
- The simple path from vertex 1 to vertex 5 is $(1, 2, 4, 5)$, and the length of the longest increasing subsequence of $(P_1, P_2, P_4, P_5) = (3, 2, 4, 5)$ is 3. Thus, $L_5 = 3$.
- From the above, $f(P) = 1 + 1 + 1 + 2 + 3 = 8$.

Hence, the permutation P in the sample output satisfies the condition $f(P) = 8$. Besides, $P = (3, 2, 4, 5, 1)$ also satisfies the condition, for example.

Sample Input 2

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

Sample Output 2

```
No
```

It can be proved that no permutation P satisfies $f(P) = 21$.

Sample Input 3

```
8 20
3 1
3 8
7 1
7 5
3 2
6 5
4 7
```

Sample Output 3

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

E - Three View Drawing

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score: 800 points

Problem Statement

> Divide a cube with a side length of N into N^3 smaller cubes, each with a side length of 1, and select K of these smaller cubes. Construct one way to select them so that, when viewed from any of the three directions perpendicular to the faces of the cubes, all K selected cubes are visible and appear in the same shape.

To formulate the problem precisely, we associate each smaller cube after division with a triple of integers (x_i, y_i, z_i).

Construct and print one set of K triples of integers (x_i, y_i, z_i) that satisfy the following conditions.

- $0 \leq x_i, y_i, z_i < N$
- $\{(x_i, y_i) \mid 1 \leq i \leq K\} = \{(y_i, z_i) \mid 1 \leq i \leq K\} = \{(z_i, x_i) \mid 1 \leq i \leq K\}$
- The set mentioned in the previous item has K elements. That is, $(x_i, y_i) \neq (x_j, y_j)$ for $i \neq j$.

It can be shown that a solution exists for any input satisfying the constraints.

Constraints

- All input values are integers.
- $1 \leq N \leq 500$
- $1 \leq K \leq N^2$

Input

The input is given from Standard Input in the following format:

```
N K
```

Output

Print your answer in the following format:

$$
\begin{array}{lll}
x_1 & y_1 & z_1 \\
x_2 & y_2 & z_2 \\
\vdots & & \\
x_K & y_K & z_K
\end{array}
$$

If multiple solutions exist, any of them will be accepted.

Sample Input 1

```
3 3
```

Sample Output 1

```
0 0 0
1 1 1
2 2 2
```

Sample Input 2

```
2 4
```

Sample Output 2

```
0 0 1
0 1 0
1 0 0
1 1 1
```

Sample Input 3

```
1 1
```

Sample Output 3

```
0 0 0
```

F - Append Same Characters

Time Limit: 4 sec / Memory Limit: 1024 MiB

Score: 1000 points

Problem Statement

You are given N strings $S_1, \ldots, S_N$ consisting of lowercase English letters. Consider performing the following two types of operations zero or more times in any order:

- Choose one lowercase letter c. Append c to the end of S_i for all $1 \le i \le N$.
- Choose an integer i such that $1 \le i \le N-1$. Swap S_i and S_{i+1}.

Find the minimum total number of operations required to make $S_i \le S_{i+1}$ in lexicographical order for all $1 \le i \le N-1$.

▶ What is lexicographical order?

Constraints

- All input values are integers.
- $2 \le N \le 3 \times 10^5$
- S_i is a string consisting of lowercase English letters.
- $1 \le |S_i|$
- $|S_1| + |S_2| + \cdots + |S_N| \le 3 \times 10^5$

Input

The input is given from Standard Input in the following format:

$$
\begin{aligned}
&N \\
&S_1 \\
&S_2 \\
&\vdots \\
&S_N
\end{aligned}
$$

Output

Print the answer in a single line.

Sample Input 1

```
5
ab
rac
a
dab
ra
```

Sample Output 1

```
3
```

Here is one way to operate.

- Swap S_2 and S_3. Now $(S_1, \ldots, S_5) = ($ab, a, rac, dab, ra$)$.
- Append z to the end of each string. Now $(S_1, \ldots, S_5) = ($abz, az, racz, dabz, raz$)$.
- Swap S_3 and S_4. Now $(S_1, \ldots, S_5) = ($abz, az, dabz, racz, raz$)$. At this point, we have $S_i \leq S_{i+1}$ for all $i = 1, \ldots, N-1$.

It is impossible to make $S_i \leq S_{i+1}$ for all $i = 1, \ldots, N-1$ with fewer than three operations, so you should print 3.

Sample Input 2

```
3
kitekuma
nok
zkou
```

Sample Output 2

```
0
```

Before any operation is performed, we have $S_i \leq S_{i+1}$ for all $i = 1, \ldots, N-1$.

Sample Input 3

```
31
arc
arrc
rc
rac
a
rc
aara
ra
caac
cr
carr
rrra
ac
r
ccr
a
c
aa
acc
rar
r
c
r
a
r
rc
a
r
rc
cr
c
```

Sample Output 3

```
175
```

Note that we may have $S_i = S_j$ for $i \neq j$.