Problem H. Parenthesis Checking

Time limit 3000 ms
Mem limit 1048576 kB

Problem Statement

Let us define a **correct parenthesis sequence** as a string that satisfies one of the following conditions.

- It is an empty string.
- It is a concatenation of `(`, A, `)`, in this order, for some **correct parenthesis sequence** A.
- It is a concatenation of A, B, in this order, for some **correct parenthesis sequences** A and B.

We have a string S of length N consisting of `(` and `)`.

Given Q queries $\mathrm{Query}_1, \mathrm{Query}_2, \ldots, \mathrm{Query}_Q$, process them in order. There are two kinds of queries, with the following formats and content.

- `1 l r`: Swap the l-th and r-th characters of S.
- `2 l r`: Determine whether the contiguous substring from the l-th through the r-th character is a **correct parenthesis sequence**.

Constraints

- $1 \leq N, Q \leq 2 \times 10^5$
- S is a string of length N consisting of `(` and `)`.
- $1 \leq l < r \leq N$
- N, Q, l, r are all integers.
- Each query is in the format `1 l r` or `2 l r`.
- There is at least one query in the format `2 l r`.

Input

Input is given from Standard Input in the following format:

N Q
S
Query_1
Query_2
$\vdots$
Query_Q

Output

For each query in the format `2 l r`, print `Yes` if the contiguous substring is a **correct parenthesis sequence**, and `No` otherwise, followed by a newline.

Sample 1

Input	Output
5 3 (())(2 1 4 2 1 2 2 4 5	Yes No No

In the first query, `(())` is a **correct parenthesis sequence**.

In the second query, `((` is not a **correct parenthesis sequence**.

In the third query, `)(` is not a **correct parenthesis sequence**.

Sample 2

Input	Output
5 3 (())(2 1 4 1 1 4 2 1 4	Yes No

In the first query, `(())` is a **correct parenthesis sequence**.

In the second query, S becomes `)()((`.

In the third query, `)()(` is not a **correct parenthesis sequence**.

Sample 3

Input	Output
8 8 ((()(())) 2 2 7 2 2 8 1 2 5 2 3 4 1 3 4 1 3 5 1 1 4 1 6 8	Yes No No