

Description

A string is said to be valid if it satisfies one of the following rules:

- (1) The string is an empty string.
- (2) If strings S_1 and S_2 are both valid, then S_1S_2 is valid.
- (3) If a string S is valid, then $\{S\}$, $[S]$, (S) and $\langle S \rangle$ are valid as long as S does not contain any parentheses with higher precedence than its enclosing parentheses. The precedence order of parentheses from high to low is $\{ \}$, $[]$, $()$, $\langle \rangle$. Therefore, a string of $\{[]()\}$ is valid, but a string of $\{([[]])\}$ is not.

Given a string consisting of parentheses, determine if it is a valid string.

Input

The first line of the input contains an integer N ($N \leq 10000$) denoting the number of test cases. Each of the next N lines corresponds to a test case, which contains a string consisting of parentheses, and the maximum string length is no more than 1000 characters.

Note that an empty string (a line which contains the newline character only) may be present in the input and it should be considered as a valid string according to rule (1).

Subtask 1: $N \leq 50$

Subtask 2: $N \leq 100$

Subtask 3: $N \leq 1000$

Subtask 4: $N \leq 10000$

Output

For each test case, print "Yes" or "No" to indicate that the string is valid or not and a newline in the end of each line.

Sample Input

```
{[()]}()  
[(<)]  
<[({})]>  
({}){ }[<>]
```

EOF

Sample Output

Yes

No

No

Yes

EOF