

Contest Duration: 2025-10-18(Sat) 23:30 (<http://www.timeanddate.com/worldclock/fixedtime.html?iso=20251018T2130&p1=248>) - 2025-10-19(Sun) 01:10 (<http://www.timeanddate.com/worldclock/fixedtime.html?iso=20251018T2130&p1=248>) (local time) (100 minutes)

iso=20251018T2130&p1=248) - 2025-10-19(Sun) 01:10 (<http://www.timeanddate.com/worldclock/fixedtime.html?iso=20251018T2130&p1=248>) (local time) (100 minutes)

[Back to Home \(/home\)](#)

[🏠 Top \(/contests/abc428\)](#)

[📋 Tasks \(/contests/abc428/tasks\)](#)

[❓ Clarifications \(/contests/abc428/clarifications\)](#)

[📊 Results ▼](#)

[⬇️ Standings \(/contests/abc428/standings\)](#)

[⬇️ Virtual Standings \(/contests/abc428/standings/virtual\)](#)

[📖 Editorial \(/contests/abc428/editorial\)](#)

[💬 Discuss \(https://codeforces.com/blog/entry/147512\)](https://codeforces.com/blog/entry/147512)



# C - Brackets Stack Query

[Editorial \(/contests/abc428/tasks/abc428\\_c/editorial\)](#)



Time Limit: 3 sec / Memory Limit: 1024 MiB

Score : 300 points

## Problem Statement

A string  $T$  is called a good bracket sequence if and only if it satisfies the following condition:

- $T$  can be made into an empty string by repeating the following operation zero or more times:
  - Choose  $()$  contained in  $T$  as a (contiguous) substring and remove it.

For example,  $()$ ,  $((()))$ , and the empty string are good bracket sequences, but  $)() ($  and  $)))$  are not good bracket sequences.

There is a string  $S$ . Initially,  $S$  is an empty string.

Process  $Q$  queries in the order they are given. After each query, determine whether  $S$  is a good bracket sequence.

There are two types of queries:

- 1  $c$ : A character  $c$  is given.  $c$  is either  $($  or  $)$ . Append  $c$  to the end of  $S$ .
- 2: Remove the last character of  $S$ . It is guaranteed that  $S$  is not an empty string at this time.

## Constraints

- $1 \leq Q \leq 8 \times 10^5$
- $c$  in queries of the first type is  $($  or  $)$ .

2026-01-02 (Fri)  
05:31:45 +11:00

- It is guaranteed that  $S$  is not empty when a query of the second type is given.
- $Q$  is an integer.

## Input

The input is given from Standard Input in the following format, where  $\text{query}_i$  denotes the  $i$ -th query.

```
Q
query1
query2
⋮
queryQ
```

Each query is given in one of the following two formats:

```
1 c
```

```
2
```

## Output

Output  $Q$  lines. The  $i$ -th line should contain **Yes** if the string  $S$  immediately after processing the  $i$ -th query is a good bracket sequence, and **No** otherwise.

### Sample Input 1

[Copy](#)

```
8
1 (
2
1 (
1 )
2
1 (
1 )
1 )
```

[Copy](#)

### Sample Output 1

[Copy](#)

No  
Yes  
No  
Yes  
No  
No  
No  
Yes

$S$  immediately after processing the 1st query is  $()$ , which is not a good bracket sequence.

$S$  immediately after processing the 2nd query is an empty string, which is a good bracket sequence.

$S$  immediately after processing the 3rd query is  $()$ , which is not a good bracket sequence.

$S$  immediately after processing the 4th query is  $()$ , which is a good bracket sequence.

$S$  immediately after processing the 5th query is  $()$ , which is not a good bracket sequence.

$S$  immediately after processing the 6th query is  $(($ , which is not a good bracket sequence.

$S$  immediately after processing the 7th query is  $(($ , which is not a good bracket sequence.

$S$  immediately after processing the 8th query is  $(())$ , which is a good bracket sequence.

'#telegram)

url=https%3A%2F%2Fatcoder.jp%2Fcontests%2Fabc428%2Ftasks%2Fabc428\_c%3Flang%3Den&title=C%20-

[Rule \(/contests/abc428/rules\)](/contests/abc428/rules) [Glossary \(/contests/abc428/glossary\)](/contests/abc428/glossary)

[Terms of service \(/tos\)](/tos) [Privacy Policy \(/privacy\)](/privacy) [Information Protection Policy \(/personal\)](/personal) [Company \(/company\)](/company)

[FAQ \(/faq\)](/faq) [Contact \(/contact\)](/contact)

Copyright Since 2012 ©AtCoder Inc. (<http://atcoder.co.jp>) All rights reserved.