



била показана на 5 юли 2018 02:35:15 GMT. [Настоящата страница](#) може да е била променена междувременно. [Научете повече](#).

[Пълна версия](#) [Версия само с текст](#) [Преглед на източника](#)

Съвет: за да намерите бързо думата си за търсене на тази страница, натиснете **Ctrl+F** или **⌘-F** (Mac) и използвайте лентата за търсене.

Validate!

[All submissions](#)

[Best submissions](#)

[Read editorial](#)

✓ **Points:** 100 (partial)

⌚ **Time limit:** 0.3s

📄 **Memory limit:** 64M

✍ **Author:**

[doncho](#)

🏷 **Tags**

Recursion

⬆ **Difficulty**

Intermediate

▼ **Allowed languages**

java

You are given parenthesis. A lot of them. And wildcards (*). Each wildcard can be replaced by opening or closing parenthesis or can be removed. Your task is to say 'yes' or 'no' if an expression of parenthesis is valid.

An expression is valid, if:

- The count of opening and closing parenthesis is equal
- Each closing parenthesis must have a corresponding opening parenthesis
- Each opening parenthesis must have a corresponding closing parenthesis
- Opening parenthesis must be left of closing parenthesis

For more clarifications, see the examples below

Input

Read from the standard input

- On the first line, read the number N
 - This is the number of expressions to follow
- On each of the next N lines, find an expression of parenthesis and wildcards

Output

Print to the standard output

- On the N lines, print either 'yes' or 'no' if the expression is valid

Constraints

- $2 \leq N \leq 10$
- The length of each expression will be at most 10

Sample tests

Input

Copy

```
2
()()
(*)
```

Output

```
yes
yes
```

[Copy](#)

Input

```
9
()()()()()
) (
(( ( **
*

***
**
( ** )
**** ) )
```

[Copy](#)

Output

```
yes
no
no
yes
yes
yes
yes
yes
yes
yes
```

[Copy](#)

Notes

Line 5 (the empty string) is a valid paranthesis expression.

Comments

There are no comments at the moment.