Balanced Expressions

Implement a program in C++ that that determines whether an expression of parentheses, “(“ or “)”, square brackets, “[“ or “]”, and curly brackets, “{“, “}” is balanced.

An expression is balanced when a closing symbol, “)”, “]” or “}”, matches the last corresponding open symbol, respectively “(“, “[“, “{”, and all the open symbols are closed at the end of the expression. Notice that, when we refer to the last open symbol, we ignore all the part of the expression that has been properly closed. Some examples of balanced expressions follow:

()

([{}()])

[][]({})

Some examples of not balanced expressions follow:

(

([]

([)]

{]

{(})()

]

Use the most appropriate linear data structure to solve this problem. You can either use a predefined linear data structure as introduced in class or use the implementation provided in the virtual campus.

Input

The first line will indicate the number of cases. Each case will be defined with a line with an expression of symbols without spaces.

Output

The output of each case should be printed in one line. The output will be “yes” if the expression is balanced or “no” otherwise.

Example of input

```
9
()
(
([{}()])
([)]
]
[][]({})
 ([]
{]
{(})()
```

Example of output

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