



Stacks: Balanced Brackets

by [saikiran9194](#)

Problem

Submissions

Leaderboard

Discussions

Editorial

Check out the resources on the page's right side to learn more about stacks. The video tutorial is by Gayle Laakmann McDowell, author of the best-selling interview book [Cracking the Coding Interview](#).

A bracket is considered to be any one of the following characters: (,) , { , } , [, or] .

Two brackets are considered to be a *matched pair* if the an opening bracket (i.e., (, [, or {) occurs to the left of a closing bracket (i.e.,) ,] , or }) of the *exact same type*. There are three types of matched pairs of brackets: [] , { } , and () .

A matching pair of brackets is *not balanced* if the set of brackets it encloses are not matched. For example, { [()] } is not balanced because the contents in between { and } are not balanced. The pair of square brackets encloses a single, unbalanced opening bracket, (, and the pair of parentheses encloses a single, unbalanced closing square bracket,] .

By this logic, we say a sequence of brackets is considered to be *balanced* if the following conditions are met:

- It contains no unmatched brackets.
- The subset of brackets enclosed within the confines of a matched pair of brackets is also a matched pair of brackets.

Given n strings of brackets, determine whether each sequence of brackets is balanced. If a string is balanced, print YES on a new line; otherwise, print NO on a new line.

Input Format

The first line contains a single integer, n , denoting the number of strings.

Each line i of the n subsequent lines consists of a single string, s , denoting a sequence of brackets.

Constraints

- $1 \leq n \leq 10^3$
- $1 \leq \text{length}(s) \leq 10^3$, where $\text{length}(s)$ is the length of the sequence.
- Each character in the sequence will be a bracket (i.e., { , } , (,) , [, and]).

Output Format

For each string, print whether or not the string of brackets is balanced on a new line. If the brackets are *balanced*, print YES ; otherwise, print NO .

Sample Input

```
3
{[(())}
{[(())}
{[[[(())]]}
```

Sample Output

```
YES
NO
YES
```

Explanation

1. The string `{[()]}` meets both criteria for being a balanced string, so we print `YES` on a new line.
2. The string `{[(())]}` is not balanced, because the brackets enclosed by the matched pairs `[(]` and `()` are not balanced. Thus, we print `NO` on a new line.
3. The string `{{[[(())]]}}` meets both criteria for being a balanced string, so we print `YES` on a new line.

[f](#) [t](#) [in](#)

Submissions: 21232

Max Score: 30

Difficulty: Medium

Rate This Challenge:

☆☆☆☆

Need Help?

5:46

[Stacks](#)[More](#)Current Buffer (saved locally, editable)  

Java 8



```
1 import java.io.*;
2 import java.util.*;
3 import java.text.*;
4 import java.math.*;
5 import java.util.regex.*;
6
7 public class Solution {
8
9     public static boolean isBalanced(String expression) {
10
11         // Must be even
12         if ((expression.length() & 1) == 1)
13             return false;
14         else {
15             char[] brackets = expression.toCharArray();
16             Stack<Character> s = new Stack<>();
17             for (char bracket : brackets)
18                 switch (bracket) {
19                     case '{': s.push('}'); break;
20                     case '(': s.push(')'); break;
21                     case '[': s.push(']'); break;
22                     default :
23                         if (s.empty() || bracket != s.peek())
24                             return false;
25                         s.pop();
26                 }
27             return s.empty();
28         }
29     }
30
31     public static void main(String[] args) {
32         Scanner in = new Scanner(System.in);
33         int t = in.nextInt();
34         for (int a0 = 0; a0 < t; a0++) {
```

```
35         String expression = in.next();
36         System.out.println( (isBalanced(expression)) ? "YES" : "NO" );
37     }
38 }
39 }
40 }
```

Line: 40 Col: 1

 [Upload Code as File](#)☐ Test against custom input

Run Code

Submit Code

Congrats, you solved this challenge!

✓ Test Case #0
✓ Test Case #3
✓ Test Case #6
✓ Test Case #9
✓ Test Case #12
✓ Test Case #15
✓ Test Case #18

✓ Test Case #1
✓ Test Case #4
✓ Test Case #7
✓ Test Case #10
✓ Test Case #13
✓ Test Case #16

✓ Test Case #2
✓ Test Case #5
✓ Test Case #8
✓ Test Case #11
✓ Test Case #14
✓ Test Case #17

[Next Challenge](#)

Copyright © 2017 HackerRank. All Rights Reserved

Join us on IRC at [#hackerrank](#) on freenode for hugs or bugs.

[Contest Calendar](#) | [Interview Prep](#) | [Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#) | [Request a Feature](#)