Balanced Brackets

locked

Submitted a few seconds ago • Score: 1.00

Status: Accepted

✔ Test Case #0	✔ Test Case #1	✔ Test Case #2
✔ Test Case #3	✔ Test Case #4	✔ Test Case #5
✔ Test Case #6	✔ Test Case #7	✔ Test Case #8
✔ Test Case #9	✔ Test Case #10	✔ Test Case #11
✔ Test Case #12	✔ Test Case #13	✔ Test Case #14
✔ Test Case #15	✔ Test Case #16	✔ Test Case #17
✔ Test Case #18	✔ Test Case #19	✔ Test Case #20

Submitted Code

Language: C++

Open in editor

```cpp
#include<string.h>
#include<stdio.h>
// #include<iostream.h>
#include<bits/stdc++.h>
using namespace std;

bool is_balanced(string expression) {
    vector<char> s; // Use a stack to keep track of opening brackets
    for(int i = 0; i < expression.size(); i++) {
        if(expression[i] == '{' || expression[i] == '[' || expression[i] == '(') {
            s.push_back(expression[i]);
        } else {
            // Pop the top of the stack and make sure the brackets match
            if(s.size() == 0) {
                return false;
            }
            char c = s.back();
            s.pop_back();
            if(c == '{' && expression[i] != '}') {
                return false;
            }
            if(c == '(' && expression[i] != ')') {
                return false;
            }
            if(c == '[' && expression[i] != ']') {
                return false;
            }
        }
    }
    if(s.size() == 0) {
        return true;
```

```
    } else {
        return false;
    }
}

int main(){
    int t;
    cin >> t;
    for(int a0 = 0; a0 < t; a0++){
        string expression;
        cin >> expression;
        bool answer = is_balanced(expression);
        if(answer)
            cout << "YES\n";
        else cout << "NO\n";
    }
    return 0;
}
```