



# Valid Expression

Problem

Submissions

Leaderboard

Discussions

Given a string  $S$ , composed of different combinations of '(', ')', '{', '}', '[', ']', ' '. The task is to verify the validity of the arrangement.

Note: Ignore the precedence of brackets.

## Example 1:

Input:  $S = ()\{\}$

Output: 1 Explanation: The arrangement is valid.

## Example 2: Input:

$S = ())(\{\}$

Output: 0

Explanation: Arrangement is not valid.

You don't need to read input or print anything. Complete the function `valid()` which takes  $S$  as input and returns a boolean value denoting whether the arrangement is valid or not.

## Input Format

You use one of the following templates to write your code.

## C template:

```
bool validString(char *s){  
  
}
```

## C++ template:

```
bool validString(string s){  
  
}
```

## Constraints

$$1 \leq N \leq 10^4$$

## Output Format

just follow the description

Contest ends in 2 hours

Submissions: 0

Max Score: 2

Difficulty: Medium

Rate This Challenge:



[More](#)

C



```
1 #include <stdio.h>
2 #include <string.h>
3 #include <math.h>
4 #include <stdlib.h>
5
6 int main() {
7
8     /* Enter your code here. Read input from STDIN. Print output to STDOUT */
9     return 0;
10 }
11
```

Line: 1 Col: 1

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

Run Code

Submit Code