

20 有效括号

Label: 栈

给定一个只包括 '(' , ')' , '{' , '}' , '[' , ']' 的字符串 s ，判断字符串是否有效。

有效字符串需满足：

左括号必须用相同类型的右括号闭合。

左括号必须以正确的顺序闭合。

输入: s = "O[]{}"

输出: true

输入: s = "([)]"

输出: false

输入: s = "{[]}"

输出: true

- 栈 (进行时)

```
class Solution {
    public boolean isValid(String s) {
        int n = s.length();

        // 特殊判断
        if (n % 2 == 1) return false;

        // 先在外部定义一个映射 pair
        Map<Character, Character> pairs = new HashMap<Character, Character>() {{
            put(')', '(');
            put(']', '[');
            put('}', '{');
        }};
        Stack<Character> stack = new Stack<Character>();
        char[] array = s.toCharArray();

        for (int i = 0; i < n; i++) {

            if (pairs.containsKey(array[i])) { // 如果 右括号 出现，则进行判断

                if (stack.isEmpty() || stack.peek() != pairs.get(array[i])) {
                    return false;
                }
                stack.pop();
            } else { // 如果 左括号 出现，则入栈
                stack.push(array[i]);
            }
        }
        return stack.isEmpty();
    }
}
```

- 数组模拟栈 (用时优化)

```
class Solution {  
  
    public boolean isValid(String s) {  
  
        if (s.length() % 2 == 1) return false;  
  
        char[] cache = new char[s.length()];  
        int index = -1;  
        char[] array = s.toCharArray();  
  
        for (char c : array) {  
            if (c=='{' || c=='(' || c=='[') { // 左括号  
                cache[++index] = c; // 数组存储  
            } else {  
                if (index < 0) {  
                    return false;  
                }  
  
                char last = cache[index]; // 栈顶  
                index--;  
  
                if (c == '}' && last != '{') {  
                    return false;  
                }  
                if (c == ')' && last != '(') {  
                    return false;  
                }  
                if (c == ']' && last != '[') {  
                    return false;  
                }  
            }  
        }  
        return index < 0;  
    }  
}
```