

## Longest valid Parentheses

Difficulty: **Hard**Accuracy: **26.13%**Submissions: **146K+**Points: **8**Average Time: **20m**

Given a string **s** consisting of opening and closing parenthesis '(' and ')'. Find the length of the longest valid parenthesis substring.

A parenthesis string is valid if:

- For every opening parenthesis, there is a closing parenthesis.
- The closing parenthesis must be after its opening parenthesis.

Examples :

**Input:** **s** = "((()"

**Output:** 2

**Explanation:** The longest valid parenthesis substring is "()".

**Input:** **s** = ")()()"

**Output:** 4

**Explanation:** The longest valid parenthesis substring is "()()".

**Input:** **s** = "())()"

```
1 // } Driver Code Ends
22 class Solution {
23     static int maxLength(String s) {
24         Stack<Integer> st = new Stack<>();
25         st.push(-1);
26         int maxLen = 0;
27         for(int i = 0 ; i<s.length(); i++){
28             if(s.charAt(i) == '('){
29                 st.push(i);
30             }else{
31                 st.pop();
32                 if(st.empty()){
33                     st.push(i);
34                 }else{
35                     maxLen = Math.max(maxLen, i - st.peek());
36                 }
37             }
38         }
39         return maxLen;
40     }
41 }
```

