



PRACTICE

COMPETE

JOBS

LEADERBOARD

Search



swatantragoswam1 ▾

[All Contests](#) > [GCFL_3_year_6_sem](#) > Valid Substring (ST2)

Valid Substring (ST2)

locked

Problem

Submissions

Leaderboard

Discussions

Editorial

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

Submitted Code

Language: C++

Open in editor

```
1 /*  
2  Link: http://www.geeksforgeeks.org/length-of-the-longest-valid-substring/  
3  Length of the longest valid substring
```

[Privacy](#) - [Terms](#)

```
4      Given a string consisting of opening and closing parenthesis, find length of the longest valid
      parenthesis substring.
5      Examples:
6      Input : (((
7      Output : 2
8      Explanation : ()
9      Input: )()(())
10     Output : 4
11     Explanation: ()()
12     Input:  ()((())))
13     Output: 6
14     Explanation:  ()((()))
15 */
16
17
18
19
20
21
22
23
24
25 #include <cmath>
26 #include <cstdio>
27 #include <vector>
28 #include <iostream>
29 #include <algorithm>
30 #include <stack>
31 using namespace std;
32
33 // Shortcuts for "common" data types in contests
34 typedef long long ll;
35 #define inf 0x3f3f3f3f
36 #define Inf 0x3FFFFFFFFFFFFFFFFFLL
37
```

```
38 // To simplify repetitions/loops, Note: define your loop style and stick with it!
39 #define rep(i, n) for (int i = 0; i < (n); ++i)
40 #define Rep(i, n) for (int i = 1; i <= (n); ++i)
41 #define REP(i, j, n) for (int i = (j); i <= (n); ++i)
42 #define clr(x) memset((x), 0, sizeof (x))
43
44 using namespace std;
45
46
47 int findMaxLen(string str)
48 {
49     int n = str.length();
50
51     // Create a stack and push -1 as initial index to it.
52     stack<int> stk;
53     stk.push(-1);
54
55     // Initialize result
56     int result = 0;
57
58     // Traverse all characters of given string
59     for (int i=0; i<n; i++)
60     {
61         // If opening bracket, push index of it
62         if (str[i] == '(')
63             stk.push(i);
64
65         else // If closing bracket, i.e.,str[i] = ')'
66         {
67             // Pop the previous opening bracket's index
68             stk.pop();
69
70             // Check if this length formed with base of
71             // current valid substring is more than max
72             // so far
```

```
73         if (!stk.empty())
74             result = max(result, i - stk.top());
75
76         // If stack is empty. push current index as
77         // base for next valid substring (if any)
78         else stk.push(i);
79     }
80 }
81
82 return result;
83 }
84
85 // Driver program
86 int main()
87 {
88     string str ;
89     cin>>str;
90     cout << findMaxLen(str) << endl;
91
92
93
94     return 0;
95 }
```