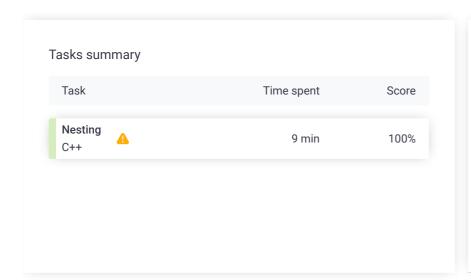
Codility_

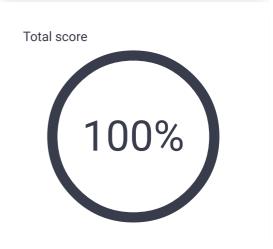
CodeCheck Report: training4JU5ZC-X5B

Test Name:

Check out Codility training tasks

Summary Timeline





Tasks Details

1. Nesting

Determine whether a given string of parentheses (single type) is properly nested.



Task description

A string S consisting of N characters is called properly nested if:

- · S is empty;
- S has the form "(U)" where U is a properly nested string:
- S has the form "VW" where V and W are properly nested strings.

For example, string "(()(()))" is properly nested but string " ())" isn't.

Write a function:

int solution(string &S);

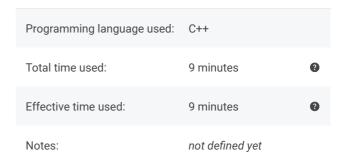
that, given a string S consisting of N characters, returns 1 if string S is properly nested and 0 otherwise.

For example, given S = "(()(())())", the function should return 1 and given S = "())", the function should return 0, as explained above.

Write an efficient algorithm for the following assumptions:

- N is an integer within the range [0..1,000,000];
- string S consists only of the characters "(" and/or ") ".

Solution





00:20:55 00:29:02 Code: 00:29:01 UTC, cpp, show code in pop-up final, score: 100 // you can use includes, for example: 2

// #include <algorithm>

3

Test results - Codility

Copyright 2009–2021 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

```
// you can write to stdout for debugging purpo
// cout << "this is a debug message" << endl;</pre>
 5
 6
 7
     int solution(string &S) {
 8
         int cont=0, n=S.size();
 9
         for(int i=0;i<n;i++){</pre>
              if(S[i]=='(' || S[i]=='{' || S[i]=='[']
10
11
                   cont++;
12
              else if(S[i]==')' || S[i]=='}' || S[i]:
13
14
                   cont --;
15
                   if(cont<0){</pre>
16
                        return 0;
17
                   }
18
              }
19
         }
20
         if(cont!=0){return 0;}
21
         else{return 1;}
22
```

Analysis summary

The solution obtained perfect score.

Analysis

Detected time complexity: O(N)

ехра	nd all Example test	s	
•	example1 example test	1	OK
•	example2 example test2	•	OK
ехра	nd all Correctness te	sts	•
•	negative_match invalid structure, but the number of parentheses matches	•	OK
•	empty empty string	•	OK
•	simple_grouped simple grouped positive and negative test, length=22	•	OK
•	small random	1	OK
	- · - · - · · · · · · · · · · · · · · ·		
ехра	nd all Performance te	sts	5
expa	nd all Performance to large1 simple large positive and negative test, 10K or 10K+1 ('s followed by 10K)'s		ok
	large1 simple large positive and negative test,	1	•
>	large1 simple large positive and negative test, 10K or 10K+1 ('s followed by 10K)'s large_full_ternary_tree tree of the form T=(TTT) and depth 11,	✓ ✓	ОК