# Codility\_

## CodeCheck Report: training4JU5ZC-X5B

Test Name:

Check out Codility training tasks

Summary

Timeline

### Mail Status: Not Applicable

Finished: 2021-10-22 00:29 UTC

Started: 2021-10-22 00:20 UTC

Invitation Created: 2021-10-22 00:20 UTC

#### Tasks Details

#### 1. Nesting

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

Task Score Correctness 100%

Performance

100%

100%

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.

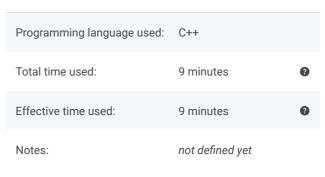
For example, given S = "(()(())())", the function should return 1 and given S = "())", the function should return 0, as explained above.

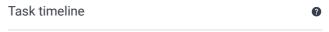
Write an efficient algorithm for the following assumptions:

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

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

#### 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:
     // #include <algorithm>
 4
     // you can write to stdout for debugging purpo
 5
     // cout << "this is a debug message" << endl;</pre>
 6
     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
12
13
             else if(S[i]==')' || S[i]=='}' || S[i]:
```

## Analysis summary

The solution obtained perfect score.

## Analysis

Detected time complexity: O(N)

ехра	nd all <b>Example test</b>	s	
<b>&gt;</b>	example1 example test	1	ОК
•	example2 example test2	•	ОК
ехра	nd all Correctness te	sts	5
•	negative_match invalid structure, but the number of parentheses matches	•	OK
•	empty empty string	1	ОК
•	simple_grouped simple grouped positive and negative test, length=22	1	OK
<b>•</b>	small_random	•	OK
ехра	nd all Performance to	est	s
•	large1 simple large positive and negative test, 10K or 10K+1 ('s followed by 10K)'s	•	OK
•	large_full_ternary_tree tree of the form T=(TTT) and depth 11, length=177K+	1	OK
•	multiple_full_binary_trees sequence of full trees of the form T= (TT), depths [1101], with/without unmatched ')' at the end, length=49K+	•	ок
<b>&gt;</b>	broad_tree_with_deep_paths	1	ОК