2.Medium\02.Max_nesting_depth_of_parenthesis.cpp

```cpp
/*
Question:
Given a VPS represented as a string s, return the nesting depth of s.

Approach:
1. Initialize `opened` as 0 and `ans` as 0 to keep track of the number of opened parentheses and the maximum nesting depth respectively.
2. Iterate through each character `c` in the string `s`.
    a. If `c` is an opening parenthesis '(', increment `opened` by 1 and update `ans` if it is greater than the current value of `ans`.
    b. If `c` is a closing parenthesis ')', decrement `opened` by 1.
3. Return `ans` as the maximum nesting depth.

CODE:-

*/

int maxDepth(string s) {
    int opened = 0, ans = 0;
    for (auto c : s) {
        if (c == '(') {
            opened++;
            ans = max(ans, opened);
        } else if (c == ')') {
            opened--;
        }
    }
    return ans;
}

/*
Time Complexity: O(n), where n is the length of the string `s`.
Space Complexity: O(1)
*/
```