AlgoExpert Quad Layout C++ 12px Sublime Monokai 00:00:00

Run Code

```
Solution 1 Solution 2
```

Prompt

Scratchpad

```
_{\rm 1} \, // Copyright @ 2020 AlgoExpert, LLC. All rights reserved.
 3 #include <vector>
 4 #include <climits>
 5 using namespace std;
 7 // O(n^2) time | O(n^2) space
 8 int palindromePartitioningMinCuts(string s) {
       vector<vector<bool>> palindromes(s.length(), vector<bool>(s.length(), false));
10
        for (int i = 0; i < s.length(); i++) {</pre>
          palindromes[i][i] = true;
11
12
       for (int length = 2; length < s.length() + 1; length++) {
    for (int i = 0; i < s.length() - length + 1; i++) {
        int j = i + length - 1;
    }
}</pre>
13
14
15
16
             if (length == 2) {
17
               palindromes[i][j] = (s[i] == s[j]);
18
19
               palindromes[i][j] = (s[i] == s[j] \&\& palindromes[i+1][j-1]);
20
21
22
23
        vector<int> cuts(s.length(), INT_MAX);
24
        for (int i = 0; i < s.length(); i++) {</pre>
25
          if (palindromes[0][i]) {
26
             cuts[i] = 0;
27
28
29
          } else {
             cuts[i] = cuts[i - 1] + 1;
for (int j = 1; j < i; j++) {</pre>
                \textbf{if} \ (\texttt{palindromes}[\texttt{j}][\texttt{i}] \ \&\& \ \texttt{cuts}[\texttt{j} \ - \ \textbf{1}] \ + \ \textbf{1} \ < \ \texttt{cuts}[\texttt{i}]) \ \{ \\
30
31
                  cuts[i] = cuts[j - 1] + 1;
32
33
34
35
36
        return cuts[s.length() - 1];
37 }
```

Our Solution(s)

Video Explanation