

Solution 1

Solution 2

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 package main
4
5 import "math"
6
7 // O(n^2) time | O(n^2) space
8 func PalindromePartitioningMinCuts(str string) int {
9     palindromes := make([][]bool, len(str))
10    for i := range palindromes {
11        palindromes[i] = make([]bool, len(str))
12    }
13    for i := range str {
14        palindromes[i][i] = true
15    }
16    length := 2; length < len(str)+1; length++ {
17        for i := 0; i < len(str)-length+1; i++ {
18            j := i + length - 1
19            if length == 2 {
20                palindromes[i][j] = (str[i] == str[j])
21            } else {
22                palindromes[i][j] = (str[i] == str[j] && palindromes[i+1][j-1])
23            }
24        }
25    }
26    cuts := make([]int, len(str))
27    for i := range cuts {
28        cuts[i] = math.MaxInt32
29    }
30    for i := range str {
31        if palindromes[0][i] {
32            cuts[i] = 0
33        } else {
34            cuts[i] = cuts[i-1] + 1
35            for j := 1; j < i; j++ {
36                if palindromes[j][i] && cuts[j-1]+1 < cuts[i] {
37                    cuts[i] = cuts[j-1] + 1
38                }
39            }
40        }
41    }
42    return cuts[len(cuts)-1]
43 }
44
```

