```python
# Copyright © 2020 AlgoExpert, LLC. All rights reserved.

# O(n^3) time | O(n^2) space
def palindromePartitioningMinCuts(string):
    palindromes = [[False for i in string] for j in string]
    for i in range(len(string)):
        for j in range(i, len(string)):
            palindromes[i][j] = isPalindrome(string[i : j + 1])
    cuts = [float("inf") for i in string]
    for i in range(len(string)):
        if palindromes[0][i]:
            cuts[i] = 0
        else:
            cuts[i] = cuts[i - 1] + 1
            for j in range(1, i):
                if palindromes[j][i] and cuts[j - 1] + 1 < cuts[i]:
                    cuts[i] = cuts[j - 1] + 1
    return cuts[-1]


def isPalindrome(string):
    leftIdx = 0
    rightIdx = len(string) - 1
    while leftIdx < rightIdx:
        if string[leftIdx] != string[rightIdx]:
            return False
        leftIdx += 1
        rightIdx -= 1
    return True

```