```
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

using System;

public class Program {
  // O(n^2) time | O(n^2) space
  public static int PalindromePartitioningMinCuts(string str) {
    bool[,] palindromes = new bool[str.Length,str.Length];
    for (int i = 0; i < str.Length; i++) {
      for (int j = 0; j < str.Length; j++) {
        if (i == j) {
          palindromes[i,j] = true;
        } else {
          palindromes[i,j] = false;
        }
      }
    }
    for (int length = 2; length < str.Length + 1; length++) {
      for (int i = 0; i < str.Length - length + 1; i++) {
        int j = i + length - 1;
        if (length == 2) {
          palindromes[i,j] = (str[i] == str[j]);
        } else {
          palindromes[i,
            j] =
            (str[i] == str[j] && palindromes[i + 1,j - 1]);
        }
      }
    }
    int[] cuts = new int[str.Length];
    Array.Fill(cuts, Int32.MaxValue);
    for (int i = 0; i < str.Length; i++) {
      if (palindromes[0,i]) {
        cuts[i] = 0;
      } else {
        cuts[i] = cuts[i - 1] + 1;
        for (int j = 1; j < i; j++) {
          if (palindromes[j,i] && cuts[j - 1] + 1 < cuts[i]) {
            cuts[i] = cuts[j - 1] + 1;
          }
        }
      }
    }
    return cuts[str.Length - 1];
  }
}

```