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

package main

import "math"

func PalindromePartitioningMinCuts(s string) int {
  palindromes := make([][]bool, len(s))
  for i := range palindromes {
    palindromes[i] = make([]bool, len(s))
  }
  for i := range s {
    for j := i; j < len(s); j++ {
      palindromes[i][j] = isPalindrome(s[i : j+1])
    }
  }
  cuts := make([]int, len(s))
  for i := range cuts {
    cuts[i] = math.MinInt32
  }
  for i := range s {
    if palindromes[0][i] {
      cuts[i] = 0
    } else {
      cuts[i] = cuts[i-1] + 1
      for j := 1; j < i; j++ {
        if palindromes[j][i] && cuts[j-1]+1 < cuts[i] {
          cuts[i] = cuts[j-1] + 1
        }
      }
    }
  }
  return cuts[len(s)-1]
}

func isPalindrome(s string) bool {
  for i := 0; i < len(s)/2; i++ {
    if s[i] != s[len(s)-i-1] {
      return false
    }
  }
  return true
}

```