



~~3 [a,b,c]~~  
~~2 [a,b]~~  
~~2 [a]~~  
~~1 [c]~~  
~~0 [ ]~~

~~3, [a,b,c]~~  
~~helper(3, [a,b,c])~~  
~~helper(2, [a,b])~~  
~~helper(1, [a])~~  
~~helper(0, [ ])~~

def helper(self, s, pivot, path):

# base

if pivot == len(s):

self.result.append(path)

# logic

for i in range(pivot, len(s)):

# choose partition

subS = s[pivot: i + 1]

if self.isPalindrome(subS):

temp = path[:]

temp.append(subS)

self.helper(s, i + 1, temp)

~~1 [c]~~  
~~2 [a,a]~~  
~~3 [a,a,b]~~  
~~3 [c,b,a]~~

result = [ [a,b,c], [c,b,a] ]