Longest Palindrome

Given a string, determine the longest substring that is a palindrome   
  
Input: A string  
Output: A string which is the longest palindrome

# Example

Input: ”mydadlikestodriveracecars”

Output: “racecar”

# Constraints

Time Complexity: O(N2)  
Auxiliary Space Complexity: O(N).

Assume lowercase characters and no spaces.

# Solution

1. Keep a longest palindrome found variable called ‘longest’
2. Odd count palindromes:
   1. Perform a loop through the string. For each character, set that as the center of the potential palindrome
   2. Iterate out from that center until the characters on the left and right do not match
   3. Compare the current palindrome length with the length of ‘longest’ and replace if its longer.
3. Even count palindromes:
   1. Perform another loop through the string. If the next character equals the current character then iterate out from the pair.
   2. Continue comparing the left and right letter outward until it does not match.
   3. Compare the current palindrome length with the length of ‘longest’ and replace if its longer.
4. Return the longest palindrome

# Notes

Manacher's algorithm allows one to do it within O(N) time, but requires O(N) auxiliary space.  
https://en.wikipedia.org/wiki/Longest\_palindromic\_substring

# 

# Resources

http://www.geeksforgeeks.org/longest-palindrome-substring-set-1/