Longest Palindromic Substring

Question: Given a string S, find the longest palindromic substring in S. You may assume that the maximum length of S is 1000, and there exists one unique longest palindromic substring.

Solutions:

```
class Solution:
    # @param {string} s
    # @return {string}
    def longestPalindrome(self, s):
        l = len(s)
        if l <= 2:
            if (s[0] != s[l-1]): return ''
            else: return s
        result = ''
        for i in range(0,l):
            palindrome = self.SearchPalindrome(s, i, i)
            if len(palindrome) > len(result): result = palindrome
            palindrome = self.SearchPalindrome(s, i, i+1)
            if len(palindrome) > len(result): result = palindrome
        return result
    def SearchPalindrome(self, string, start, end):
        while(start>=0 and end < len(string) and string[start]==string[end]):
            start -= 1
            end += 1
        return string[start+1:end]
Solution().longestPalindrome("bananas")
```