

Introduction to Computer Science, Winter Semester 2017
Practice Assignment 6

Exercise 6-1 String Manipulation - Palindrome

Write an algorithm that determines whether the String the user inputs is a **palindrome** or not. A palindrome is a piece of text that can be read the same way in either direction (left to right and right to left). Examples of palindromes include words such as **racecar** and **noon**.

Solution:

```
x = input()
i = 0
j = len(x) - 1
palindrome = True

while(i < j and palindrome == True):
    _ if(x[i] != x[j]):
        __ palindrome = False
        _ i += 1
        _ j -= 1

if(palindrome):
    _ print("The string is a palindrome")
else:
    _ print("The string is not a palindrome")
```

Exercise 6-2 String Manipulation - Run Length

Given a String containing uppercase characters (A-Z), write an algorithm that compresses repeated 'runs' of the same character by storing the length of that run.

Example:

Input: WWWWWWWWWWWBWWWWWWWWWWBBBBWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
Output: 12W1B12W3B24W1B14W

Solution:

```
x = input()
i = 0
count = 1

while (i < (len(x) - 1)):
    _ if(x[i] == x[i+1]):
        __ count += 1
    _ else:
        __ print(count,x[i],end=" ")
        __ count = 1
```

```
    i += 1
    print(count, x[i], end=" ")
```