Write a technical brief that explains all the submitted solutions that includes both time and space complexity. Do your best to re-create an “at the whiteboard” experience. Help us understand your thought process.

**Technical Brief**

Project Title: Fun with Palindromes

Name: Marvin Abante

Date: September 16, 2021

1. CODING TEST INFO

Language: Python3

Time Frame: 72 Hours

Time Started: 9AM September 14, 2021

1. CODING DETAILS

Level One Test:

Task: Make a function that accepts a string and returns Boolean to check if it’s a palindrome.

Problem Approach:

* Create a method that would FIRST convert any string into a lower form then remove all

Punctuations and special characters so that alphanumeric values are allowed only.

Using the process, we will be able to sanitize the given data in-order to process.

* Results of the first approach could now be check on the next method checking if the given

String is a palindrome or not by REVERSING the string and compare it to the original sanitized format.

* All output is either “true” or “false” only

Files:

* palindromeChecker.py – file for checking if string is palindrome, input is directly supplied but

modification of file to accept input could refactored if desired.

* test\_palindrome.py – located at `./test/test\_palindrome.py` is used to test the script with the

help of ‘pytest’ and ‘unittest’.

Complexity:

* Since we have no loops and data is processed directly, therefore we have O(N) since it is linear.

Level 2 & 3 Test:

Task: Make a function that returns the longest palindrome substring and

Create a function that returns the minimum cut to get a substring palindrome.

Problem Approach: (Honestly I’m lost understanding Manacher’s Algorithm but with idea in mind)

* I decided to use Manacher’s Algorithm but got out of time due to its complex understanding
* Regarding to 3rd Level test my approach was using Manacher’s Algo then append valid palindrome to an array then count.

Complexity:

* The best approach which is O(N) is Manacher’s Algorithm which had taken all my time