**Winter 2016 - Cmpt 103**

**Section 41 - Lab 1 mark sheet**

**Student name :** Metehan Dagsuyu **Total: 90 marks**

**Programming style: [13 marks]**

* Variable name problems: **-1 mark**
  + You use ‘i’ as a loop index which is okay when used with range(len(…)) although better variable names could be chosen. However, you also use ‘i’ as a loop index to represent an item in the list. In this instance, i is not a good choice. Instead, use item as in ‘for item in The\_List:’ so it’s clear what it represents.
* Properly spaced code: **-2 marks**
  + Blank lines are not always used between different sections of code within the functions
* Function header and comment problems: **-4 marks**
* Refer to the ‘Lab 1 – Presentation’ in the ‘Lab 1’ folder in Blackboard for the proper format for the function headers. Yours are not written properly.
* Your function headers are **missing the proper syntax** for each function and there is **no purpose section**.
* You need to have four comments above each function (purpose, syntax, parameters, and return value) that look like this:

# Purpose: Determine length of longest string in The\_List

# Syntax: length = max\_length(The\_list)

# Parameters: The\_list: a list of strings

# Return: length: length of the longest string in the list (int)

* Writing comments for virtually every line in your code is not useful. Instead, you want to use blank lines to separate your code into different logical blocks and then use a comment to explain what the purpose of that block is. For example, you wrote:

def count\_odd(TheList):

total\_odd = 0 #sets the total number of odd ints to 0 as it might add some onto it in the future, after checking 'TheList'

for i in TheList: #for every data in 'TheList' it will do the following once

if i % 2 == 1 : #checks to see if the integer is even or odd by using the modulus function

total\_odd += 1 #if the integer is odd this line adds one to the total count of odd numbers

return total\_odd #returns 'total\_odd' which is the total number of odd integers in 'TheList'

but it could be written without as many comments like this instead:

def count\_odd(The\_List):

total\_odd = 0

# Count the amount of odd integers in 'The\_List'

for item in The\_List:

if item % 2 == 1:

total\_odd += 1

return total\_odd

* When the instructions state to return a value, your function needs to have a ‘return’ statement inside of it. For instance, if the instructions state to return the length, then the function needs to use ‘return length’ rather than ‘print length’.
* Good use of an example to illustrate what alternate does.

**Question 1: [28 marks]**

* Function max\_length is done correctly.
* Function print\_right\_aligned is done correctly.

**Question 2: [26 marks]**

* Function count\_odd is done correctly.
* Function percent\_odd :
  + Does prevent a divide by zero error if the list is empty but it prints the result rather than returning it: **-1**
  + Does not return the percentage of odd integers in the list. It prints them instead: **-1**

**Question 3: [23 marks]**

* Good use of the helper function short\_list\_finder
* Function alternate does not return the list of alternating elements. It prints them instead: **-1**