CS 171

Lab Assignment 13

This lab assignment uses many elements provided in the main bibliographic reference for

these lectures:

Programming in Python 3

A Complete Introduction to the Python Language,

2nd Edition,

Mark Summerfield

**Exercises**

**Exercise 1 Group List**

Given a list, the task is to write a Python program to group each increasing and decreasing run. This is known as a monotonous grouping. A list is monotonic if it is either monotone increasing or monotone decreasing. A list A is monotone decreasing if for all i <= j, A[i] >= A[j].

Examples:

Input: test\_list = [5, 6, 2, 9, 7, 1, 10, 4, 2, 1, 11, 12, 2]

Output: [[5, 6], [2], [9], [7, 1], [10], [4, 2, 1], [11, 12], [2]]

*Explanation: 6 > 5 and then 2 is smaller than 6, hence becomes decreasing and new group is started. 2 and 9 being peak or transit elements, belong to individual groups.*

**Exercise 2 Subtract K**

Given a list, the task is to write a Python Program to subtract K from each digit, if the element gets below 0, retain 0.

Examples:

Input: test\_list = [2345, 8786, 2478, 8664, 3568, 28], K = 4

Output: [1, 4342, 34, 4220, 124, 4]

*Explanation:* *In 2345, 4 subtracted from 2 is -2, hence ceiled to 0. Hence just 5-4 = 1, is retained and thus output.*

Input: test\_list = [2345, 8786, 2478, 8664, 3568, 28], K = 3

Output: [12, 5453, 145, 5331, 235, 5]

*Explanation:* *In 2345, 3 subtracted from 2 is -1, hence ceiled to 0. Hence just 5-3 = 2 and 4-3 = 1, are retained. and thus output.*

**Exercise 3. Check Similar Strings**

Given two strings, the task here is to write a python program that can test if they are almost similar. Similarity of strings is being checked on the criteria of frequency difference of each character which should be greater than a threshold here represented by K.

Examples:

Input: test\_str1 = ‘aabcdaa’, test\_str2 = “abbaccd”, K = 2

Output: True

*Explanation: ‘a’ occurs 4 times in str1, and 2 times in str2, 4 – 2 = 2, in range, similarly, all chars in range, hence true.*

Input: test\_str1 = ‘abcdabcdabcd’, test\_str2 = “efgh”, K = 2

Output: False

**Exercise 4. Mapping Dictionary**

Given two dictionaries with list values, perform mapping of keys of first list with values of other list, by checking values-key linkage.

Examples:

Input: test\_dict1 = {“Gfg” : [4, 10], “Best” : [8, 6], “is” : [9, 3]}, test\_dict2 = {6 : [15, 9], 8 : [6, 3], 7 : [9, 8], 9 : [10, 11]}

Output: {‘Best’: [6, 3, 15, 9], ‘is’: [10, 11]}

*Explanation:“Best” has 8 and 6, which are mapped to 6, 3 and 15, 9 hence output for that key.*

Input: test\_dict1 = {“Gfg” : [4, 10], “Best” : [18, 16], “is” : [9, 3]}, test\_dict2 = {6 : [15, 9], 8 : [6, 3], 7 : [9, 8], 9 : [10, 11]}

Output: {‘is’: [10, 11]}

*Explanation: Only 9 present as possible key.*

**Exercise 5 DataTime & Calendar**

Given a year. Your task is to find the number of every day in a year ie.number of Monday, Tuesday, Wednesday, Thrusday, Friday, Saturday, Sunday in that given year.

***Note:*** *We have to make some key observations. The first one will be that there are at least 52 weeks in a year, so every day will occur at least 52 times in a year. As 52\*7 is 364 so the day occurring on the 1st January of any year will occur 53 times and if the year is a leap year then the day on the 2nd January will also occur 53 times.*

Examples:

Input: 2019

Output: Monday-52

Tuesday-53

Wednesday-52

Thrusday-52

Friday-52

Saturday-52

Sunday-52

Input: 2024

Output: Monday-53

Tuesday-53

Wednesday-52

Thrusday-52

Friday-52

Saturday-52

Sunday-52