CSC/DSCI 2720: Data Structures Assignment 2

Due: 02/20/2023 @ 11:59 PM ET

Answer the below questions.
You may use whatever IDEs / editors you like, but you must submit your responses on iCollege as .py files. Failure to comply with this simple requirement will result in a score of Zero. Please, be careful not to be assigned a Zero score this way.

Few Rules to be followed, else will receive a score of ZERO

(1) Your submissions will work exactly as required.

(2) Your files shall not be incomplete or worse corrupted such that the file does not compile at all. Make sure you submit a file that compiles

(3) Your submission will show an output. Should you receive a Zero for no output shown do not bother to email me with "but the logic is perfect"!

Note that your program's output must exactly match the specs (design, style) given here for each problem to pass the instructor's test cases.
Design refers to how well your code is written (i.e., is it clear, efficient, and elegant), while Style refers to the readability of your code (commented, correct indentation, good variable names).

1. [30 points] Implement the function *has_duplicates_word_finder()*. Input will be given a positive integer n, then a list contains n elements that each containing a **string**. Output should be either ***True*** if there are any duplicates among these strings or ***False*** if there are not. Your code should work well on long lists of strings. For example, input "5 [spring, summer, fall, summer, winter]" will return ***True***; input "4 [spring, summer, fall, winter]" will return False. (Hint: (1) You need to program an user's input on the integer and the words (2)You might use *set* in python.)

2. [40 points] Implement the function *sort_flowers()*. Input will be given a positive integer n, then a list contains n elements that each containing a flower name (**string**) which always starts with capital letter (e.g., "Rose", but not "rose"). The program should sort the flower names in Alphabetical order. In this exercise you can assume that the capital letters of flower names are all different. You can make your choice to use any one of sorting algorithm from bubble sort, selection sort or insertion sort. For example, input "3 [Rose, Lily, Tulip]*"* will return "[Lily, Rose, Tulip]".

3. [30 points] In this problem, your task is to complete the *reverse_list()* function. You will write a function to reverse a list of integers. Your algorithm **must** use O(1) space beyond the input (any algorithms that use space not in O(1) will receive half credit at most). You may not use any library functions in this question (any solutions that do will receive zero credit). Your program should accept input as a list of integers and output another list of integers that contains the original elements in reversed order. For example, the input "[3, 4, 7, 6, 1]" should result in printing the output "[1, 6, 7, 4, 3]".

Very Very Important:

(1) Your code should be well commented which explains all the steps you are performing to solve the problem. A submission without code comments will immediately be deducted 15 points!

(2) As a comment in your code, please write your test-cases on how you would test your solution assumptions and hence your code. A submission without test cases will immediately be deducted 15 points! Example of cases to be tested for are like: What if the array input which is expected does not exist - that is, input is a null. How should your code handle such a situation? Maybe output some message like "Null input case, so no output"? What if the length of the array is one? … so on and so forth.

Please Remember: Although, written as comments - You will address your test cases in the form of code and not prose :)