Assignment may be submitted up to 24 hours late with a 25% penalty.

Note: There is no 48 hours late submission policy on this assignment.

CMPS 150
Fall 2021
Programming Assignment #7

Date Assigned: Sunday, November 21, 2021
Due Date: 11:55 PM, Monday, November 29, 2021

Objectives:
- File input, repetition statements, selection, functions, lists, formatted output

1) Include the following information as comments at the beginning of your source code. Name it **`pa7.py`** BE SURE it ***lines up*** nicely as you see it below.

```
# Author:           Type-Your-Name
# ULID:             Type-Your-ULID
# Course/Section:   CMPS 150 – Lecture Section # ___
# Assignment:       pa7
# Date Assigned:    Sunday, November 21, 2021
# Date/Time Due:    Monday, November 29, 2021 -- 11:55 pm
#
# Description:     This program reads an answer key and student answers from input
#                  files and stores the data in two lists. The solutions are
#                  compared to the key and various statistics computed.
#
# Certification of Authenticity:
# I certify that this assignment is entirely my own work.
```

2) Description

This program reads an answer key and a student's answers for True (T) / False (F) questions from two (2) separate input files and stores the data in two (2) lists. The first input file (the solutions) is called **key.py**, and your program should read this data and append it to a list. The second input file (the student's answers) is called **studentAnswers.py**, and this data should be read and appended to a second list. Both files will be the same, **unknown length**, and the sentinel value in both is the letter **'X'**. Once you've created these lists, you will process the data and display the results. The following computations must be performed:

- Compare the two lists item-by-item. For every correct answer, append the corresponding problem number to a third list (a "correct answer" list). For every wrong answer, append the corresponding problem number to a fourth list (a "wrong answer" list).
- While comparing the lists, also keep a count of the number of "T" answers the student gives, and the number of "F" answers the student gives. You will use these counts to determine if the student answered "T" or "F" the most often.
- Finally, compute the student's average and assign a letter grade based on a 10-point scale (90-100 is an A, 80-less than 90 is a B, etc.)

Output: Your program will display the following (**see sample run**):
- A list of correct answers and the number of correct answers (hint: you can use the **len** of the correct answer list)

- A list of wrong answers and the number of wrong answers
- The count of "T" answers the student gave
- The count of "F" answers the student gave
- Which letter, "T" or "F", the student used the most
- The student's average and the corresponding letter grade (10-point scale)

Note: The code to display the answer key and student solutions is given to you and must be included in your solution.

There is no formatted output for your calculations.

Functions: All code must be contained in a function (most of it will be in main()). You must include the following user-defined functions:

- A function to determine if "T" or "F" was used the most.

  Input parameters: there are two (2) input parameters

  - An integer that represents the number of "T" answers
  - An integer that represents the number of "F" answers

  Return values: The function returns one value – a string that is 'T' if true occurs the most often or 'F' if false occurs the most often.

- A function to determine the letter grade based on the student's score

  Input parameters: there is one (1) input parameter

  - An integer that represents the student's score

  Return values: The function returns one value – a string that indicates if the score is an A, B, C, D, or F.

Hint: Look at the code we wrote in chapter 6 (functions).

A user-defined function called **PrintAnswerHeader** is provided. It displays the answer key and student's answers. <u>You must use this function in you program</u>.

Input parameters:

- **keyList**: a list that contains the solutions (the answer key)
- **answerList**: a list that contains the student's answers

There are no return values.

To use the function in main():

PrintAnswerHeader (nameOfYourListWithAnswerKey, nameOfYourListWithStudentAnswers)

3) Sample Run

```
          1  2  3  4  5  6  7  8  9  10
---------------------------------------
Key       T  F  F  T  F  T  T  F  T  T
Student   T  F  T  T  F  T  F  F  T  T

CORRECT answers: # 1 2 4 5 6 8 9 10
Number correct: 8

WRONG answers: # 3 7
Number wrong: 2

The student has 6 True answers
The student has 4 False answers
The student answered T the most often.

The student's average is 80.0 %
The letter grade is: B
```

Sample run shows 10 input values for the key and student answers, but this number can vary.

4) *Upload to Moodle*

- Get in a browser and login to Moodle.
- Go to your Lecture Section on the Moodle site.
- Click on the submission link for Programming Assignment #7.
- Select to "Add a Submission" then "Upload a File"
- Select to "Choose a File" and go about the process of browsing/finding "**pa7.py**" on the computer.
- Select to "Upload this File"
- **When returned to the Upload screen, MAKE SURE to click on the "Save Changes" button.**
- You will be returned to the "Programming Assignment #7" screen.
- This time you should see your source code file listed on it.

5) *Logout of Moodle*

You can turn in programs

up to 24 hours late for a maximum of 75% credit

NOTE: There is no 48 hour submission policy on this assignment.