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 #8

Date Assigned: Sunday, November 28, 2021
Due Date: 11:55 PM, Thursday, December 2, 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 **`pa8.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:       pa8
# Date Assigned:    Sunday, November 28, 2021
# Date/Time Due:    Thursday, December 2, 2021 -- 11:55 pm
#
# Description:     This program is an extension of pa7.  It reads an answer key
#                  from an input file and stores the data in a list. It then reads
#                  a file containing strings of several students' answers. The
#                  students' answers 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. The answers are compared, various statistics are computed, and a grade is calculated.

The first input file (the solutions) is called **key.py**, and your program should read this data and append it to a list. It is a file of **unknown length** with a sentinel value of **'X'**. This step is the same as in pa7.

After creating the answer key list, you can process the second input file. The second input file is called **studentAnswers.py,** and its format differs slightly from pa7. Each line of this input file contains a string of T/F answers for a single student's test answers. For example: TFTFFTFTTT . It is a file of **unknown length** (i.e., unknown number of student test answers) with a sentinel value of **END**. You will set up the while loop, read a single line from this input file, and perform all of the processing necessary for that string. For this input file you can append the string data to a list (similar to pa7) if you choose. However, you'll find the code to process the input as a string nearly the same as processing a list, so you can also choose to leave it as a string. Inside this while loop, the following computations must be performed:

- Compare the answer key to the student's answers. 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). This is the same as pa7.

- 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. This is the same as pa7.
- 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.). This is the same as pa7.

Since you are processing multiple students tests, you will keep track of/calculate a few other statistics:

- A count of the number of entries in the input file (i.e., # of student exams)
- The highest and lowest test averages (max and min)
- Calculate the class test average (find the average of all test averages)

Output: Your program will display the following (**see sample run**):
This output is the same as pa7:

- 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.

The new results to display:

- The number of students with test scores (the number of entries in the **studentAnswers.py** input file)
- The class average
- The highest average
- The lowest average

The class, highest, and lowest averages should be displayed with **2 decimal places of precision**.

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

The functions are the same as pa7:

- 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.

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

  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)

Additional user-defined functions:

- **`def CreateStringList (theList, filename, sentinelValue):`**

  A function that reads string data from a file of unknown length and appends the string to a list (use this function to create your answer key list). You must also close the input file.

  Input parameters: there are 3 input parameters

  - The empty list that data will be appended to (a list)
  - The name of the input file to be opened and read from (a string)
  - The sentinel value (a string)

  Return values: The function returns the list with the appended data

  Optional: If you'd like to use the pass-by-reference approach, you can pass the empty list as an input parameter and not return any results.

- **`def PrintListOnSingleLine (theList):`**

  A function that displays to the screen the contents of a list on a single line (use for printing the list of correct answers and incorrect answers)

  Input parameters: there is 1 input parameter

  - The list to be printed

  Return values: There are no return parameters

3) Sample Run

```
**************************************
Student 1 Summary:

          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

**************************************
Student 2 Summary:

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

CORRECT answers: # 2 3 5 8
Number correct: 4

WRONG answers: # 1 4 6 7 9 10
Number wrong: 6

The student has 0 True answers
The student has 10 False answers
The student answered F the most often.

The student's average is 40.0 %
The letter grade is: F

**************************************
Student 3 Summary:

          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  T  F  T  T
```

```
CORRECT answers: # 1 2 4 5 6 7 8 9 10
Number correct: 9

WRONG answers: # 3
Number wrong: 1

The student has 7 True answers
The student has 3 False answers
The student answered T the most often.

The student's average is 90.0 %
The letter grade is: A

************************************
Student 4 Summary:

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

CORRECT answers: # 3 4 7 9 10
Number correct: 5

WRONG answers: # 1 2 5 6 8
Number wrong: 5

The student has 7 True answers
The student has 3 False answers
The student answered T the most often.

The student's average is 50.0 %
The letter grade is: F

************************************
   Students:   4
  Class Avg:   65.00
Highest Avg:   90.00
 Lowest Avg:   40.00
```

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 #8.
- Select to "Add a Submission" then "Upload a File"
- Select to "Choose a File" and go about the process of browsing/finding "**pa8.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 #8" 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.