CIS 41A - Lab 3: data structures, file IO, exception handling

Write a program that calculates the GPA for a group of student records and lets the user add a new student record.

**Input file**

The input text file is lab3in.txt. Each line of the file is one student record, with comma separated fields.

The first field of each record is a student name, followed by at least one grouping of 3 fields: class name, units, letter grade. There can be one or more grouping of the 3 fields.

Example of one line of the input file: Grace Hopper,CIS 22A,4.5,A+,BUS 55,5,B+

**Program requirements**

The program should have at least 6 functions as described below, but you can add more functions as you see fit.

1. A getData function:
   * Read in each line of the file and store it in a dictionary
     + key = student name
     + value = list of tuples, where each tuple is the 3 fields for one class
     + Example from the input line above: key = Grace Hopper

value = [ (CIS 22A,4.5,A+), (BUS 55,5,B+) ]

* If the file open is not successful, print an error message and end the program

1. A printData function:

* If the GPA is not calculated yet, call the calcGPA function to calculate the GPA for each student, then store the GPA for each student as you see fit.

1pt of the lab: At the top of the program, in a comment block, explain where you store the GPA for each student. (Hint: some storage locations are easier to work with than others)

* Print all student records, in alphabetical order by first name.
* Each record is printed as multiple lines of text
  + first line: student name
  + subsequent lines: class name : units, letter grade (in column format)
  + last line: the GPA

See sample output.

1. A printFile function:

* Print all data in the same format as the printData function, but to the file lab3out.txt

1. A calcGPA function:

* Calculate the GPA for one student record.
* Create a look up table with the following data:

A+ A A- B+ B B- C+ C C- D+ D D- F

4.0 4.0 3.7 3.3 3.0 2.7 2.3 2.0 1.7 1.0 0.7 0.3 0.0

* To calculate the GPA, multiply the units and the numeric grade of each class and sum up all the products, then divide by the sum of all the units.
* Example of the input line above: (4.5 \* 4.0 + 5 \* 3.3) / (4.5 + 5) = 3.63
* The GPA is printed with 2 digits after the decimal point.

1. An addData function:

* Loop to prompt the user until you have a valid student record to add to your database of student records
* Prompt for the student name and the number of classes
* For each class, prompt for a comma separated line of class name, units, grade
* Check that the number of classes is an integer, check that the units is a positive floating point number between 0.5 and 5.0, and check that the grade is one of the letter grades shown in the look up table above.
* When the prompt loop is done, add the student record to your database, after removing any extra spacing in the comma separated fields.

1. Write a main function:

* Call getData to create a database of student records from the input file
* Call printData to print student records and GPA in a nice, column format
* Call addData to add a new student record with input from the user
* Call printData again to see the updated student database
* Call printFile to save all data to an output file

**Additional requirements**

* Have a begin documentation block for your program
* Have a docstring for all functions except main

To challenge yourself:

* Use exception handling instead of an if statement when Python can raise the exception for you.
* Use comprehension when you need to create a new iterable from an existing one.

**Sample output (**User input is shown in green)

Carlos Guestrin

CIS 41A : 4.5 A

BUS 55 : 5 A

BUS 18 : 5 B+

HUMI 16 : 4 A-

GPA: 3.75

Grace Hopper

CIS 22A : 4.5 A+

BUS 55 : 5 B+

GPA: 3.63

Guido van Rossum

CIS 41A : 4.5 A-

CIS 22A : 4.5 A

ARTS 1A : 4 B-

GPA: 3.50

Ima Student

CIS 40 : 4.5 C

PE 1 : 0.5 F

GPA: 1.80

Katherine Goble

CIS 22C : 4.5 A-

MATH 1B : 5 A

POLI 1 : 5 C

GPA: 3.22

Linus Torvalds

CIS 18B : 4.5 A

BIOL 10 : 5 C

GPA: 2.95

Enter student name: A Student

Enter number of classes: one

Enter an integer for number of class

Enter number of classes: 1

Enter comma separated class name, units, grade: PE 1

Format: class name, units, grade

Enter comma separated class name, units, grade: PE 1, a, 3

Enter a float instead of a

Enter comma separated class name, units, grade: PE 1, -3.5, 3

Units must be a float and between 0.5 and 5.0

Enter comma separated class name, units, grade: PE 1, 0.5, 3

Grade must be A-F and optional + or -

Enter comma separated class name, units, grade: PE 1, 0.5, D

A Student

PE 1 : 0.5 D

GPA: 1.00

Carlos Guestrin

CIS 41A : 4.5 A

BUS 55 : 5 A

BUS 18 : 5 B+

HUMI 16 : 4 A-

GPA: 3.75

Grace Hopper

CIS 22A : 4.5 A+

BUS 55 : 5 B+

GPA: 3.63

Guido van Rossum

CIS 41A : 4.5 A-

CIS 22A : 4.5 A

ARTS 1A : 4 B-

GPA: 3.50

Ima Student

CIS 40 : 4.5 C

PE 1 : 0.5 F

GPA: 1.80

Katherine Goble

CIS 22C : 4.5 A-

MATH 1B : 5 A

POLI 1 : 5 C

GPA: 3.22

Linus Torvalds

CIS 18B : 4.5 A

BIOL 10 : 5 C

GPA: 2.95