Assignment 3 (300 pts) Submit the file online by 11:00 am on July 26, 2018

Write a Java program that prompts user for a file name, opens and reads the data, process, and store them in an array of students. Must use the 2 classes developed for assignment 2, Student and Course. The program calculates the total completed units and GPA for a student. You may want to modify Student class to include new fields, it is up to you as how to do it.
The program then prints a menu to allow user to:

1) Add a new student to the list (make sure not to add existing students)
2) Remove a student from the list
3) Sort the list alphabetically based on student last name.
4) Print the list in a tabular form to the monitor and a file(Refer to the sample below.)
5) Exit

Use the following table to assign points for a course:

Grade	Points
A	**4.0**
B	**3.0**
C	**2.0**
D	**1.0**
F	**0**

To calculate the GPA multiply units by points for each course, and then add then together and divide the sum by total units. For example if course 1 has 5 units and the grade is A (4 points) and course 2 has 4 units and the grade is B (3 points), and course 3 has 4 units and the grade is F then (4*3 + 5*4 + 4 * 0)/13 = 2.462. Units completed are all the courses that have a passing grade (a,b,c, and d), Units taken is sum of all units registered for.

Completed courses are considered the ones with passing grade (A,B,C,D).

Input File Format:

```
Last, First
ID     number_of_courses
Course 1 name
Grade Units
Course 2 name
Grade   Units
Course 3 name
Grade   Units
```

Input File Sample:

```
Smith Jr., Joe
111-22-3333 3
Physics I
A 5
English 1A
B 4
English 1B
F 4
Jones , Bill
111-11-1111  4
Physics I
A 5
Chemistry 1A
B 5
Computer Science 1
A 4
Chemistry 1 Lab
B 1
```

Requirements:

- Develop classes to represent a student and a course: (the following data members must be included, more is OK) :**Name (First, Last) , Id number, list of courses, number of units taken, units completed, units taken, and average.**
- You may include additional classes and use composition or inheritance
- Program must be modular (all tasks must be done in methods)
- Must include methods to process array of students
- Prompt user for all input and output file names
- Use data file Assign3S18.txt to test your program.
- Set array size to 20 for students and courses but when adding elements check the size to make sure not to overflow the array or use ArrayList.

Sample output file format: (Final report to the monitor and the file) :

```
Name                Id              Units Taken     Units Complete    Average
=========================================================================
Joe Smith Jr.       111-22-3333     13              9                 2.462
Bill Jones          111-11-1111     15              15                3.600
Nancy Brown         222-11-1111     42              42                4.000
-----------------------------------------------------------------------------------------------------------
Total number of Students:           3
Total Units completed by all students:  66
Total Units taken by all students:  70
```

Required files to submit are all java files and your documentations (UML and algorithms, pseudocode), you may submit the individual files or zip the NetBeans project and submit it. Minimum classes required are:

- ***Student.java***
- ***Course.java***
- ***MangeStudent.java***