COMP 1602: Computer Programming II
Assignment #2
Date Due: **February 21st, 2018 at 11:55pm**

This assignment requires you to write a program to manipulate student, course and grade sheet information using structures. There are three files, one containing student personal information, one containing course information and one containing grade information. The program should provide a menu to enable the user to perform various queries on the data.

Student File: students.txt

The *students.txt* file contains information on a set of students, one student per line. The amount of students is unknown beforehand. Each line of data consists of a student id, the student's first name, the student's last name, gender (a single character), email address, phone number (string), and his/her date of entry (in the format dd/mm/yyyy) into the university. For example,

```
81612345        John    Smith   M       jsmith@hotmail.com              789-9089        12/2/2016
81612346        Jane    Jones   F       jjones@icloud.com               324-1678        9/5/2017
81512347        Mary    Paul    F       mpaul@yahoo.com                 215-4578        23/8/2015
81512348        Paul    Kansas  M       paul_kansas@gmail.com           515-0570        17/9/2015
0
```

A student id of 0 terminates the data in the file. There are at most 100 students in the file.

Course File, courses.txt

The *courses.txt* file contains information on a set of courses, one course per line. The amount of courses is unknown beforehand. Each line of data in the file contains the course code, the name of the course and the number of credits. For example,

```
FOUN1105        Scientific_And_Technical_Writing                3
COMP1600        Introduction_to_Computing_Concepts              3
COMP1601        Computer_Programming_I                          3
INFO1600        Introduction_to_Information_Technology_Concepts  3
ACCT1002        Introduction_to_Financial_Accounting            3
END
```

A special symbol, "`END`", indicates the end of the data.

Grade Sheet File: gradesheet.txt

The *gradesheet.txt* file contains the grades obtained by students in courses (in random order). Each line of data in the file contains a student id, a course code and the grade obtained by the student in that course. For example:

```
81612345        FOUN1105        B
81612345        INFO1600        C
81612345        COMP1601        B+
81612345        ACCT1002        A-
81612345        MATH1115        F3
81612346        FOUN1005        B+
81612346        ACCT1002        A-
81512347        COMP1601        F1
0
```

COMP 1602: Computer Programming II
Assignment #2
Date Due: **February 21st, 2018 at 11:55pm**

As shown in the example, a student id may appear zero times, one time, or several times in the file. Similarly, a course code may appear zero times, one time, or several times in the file.

A student id of 0 terminates the data in the file.

Your program should read the data from *students.txt, courses.txt* and *gradesheet.txt* and store them in appropriate arrays of structs.

After reading the data, your program should present the user with a menu with the following options:

1. Course Information: Prompt the user for a course code and print to the screen all the details of the course, that is, the course code, course name and the number of credits, all appropriately labelled.

2. Course Summary Grade Sheet: Prompt the user for a course code and print to the screen a list consisting of the student id and the grade obtained by each students who did the course.

3. Student Grade Report: Prompt the user for a student id and print to the screen a list consisting of the course code(s) and grade(s) of all the courses the student has taken if the student exists.
   If the student exists but is not enrolled in any course, display "Student is not enrolled in any course."
   If the student does not exist, display an appropriate message.

4. Student Information: Prompt the user for a student id and print to the screen, all the details of the student, that is, his/her student id, first name, last name, gender, email address, phone number, date of entry into the university and current GPA, all appropriately labelled.

   A function, `calculateGPA`, must be written to calculate a student's GPA. This function is given the student's id, gradesheet array, the number of elements in the gradesheet array, the courses array, and the number of elements in the courses array as parameters. It calculates and returns the student's GPA.

   A student's GPA is the average obtained by dividing the *total grade points* earned by the *total quality hours* for which the student has registered for, in any given period of time.

   "Quality points" means the numerical value assigned to the relevant letter grade earned. Table 1 below shows the quality points that are assigned to each grade.

Grade	Mark	Quality Points
A+	90-100	4.3
A	80-89	4
A-	75 -79	3.7
B+	70-74	3.3
B	65-69	3
B	60 - 64	2.7
C+	55-59	2.3
C	50-54	2
F1	40-49	1.7
F2	30-39	1.3
F3	0-29	0

Table 1: Grading scheme with quality points

   "Quality hours" is the amount of credits for each course. Quality hours shall be assigned even when a grade of F1, F2, or F3 is obtained in a course.

Grade points are determined by multiplying the quality hours by the quality points obtained in a course.

From the sample grade sheet data above for student `81612345`, the GPA calculation is shown below.

Course Code	Credit Hours	Grade	Grade Points
FOUN1105	3	B	3 * 3 = 9
INFO1600	3	C	3 * 2 = 6
COMP1601	3	B+	3 * 3.3 = 9.9
ACCT1002	3	A-	3 * 3.7 = 11.1
MATH1115	3	F3	3 * 0 = 0

```
Total Credit Hours: 3 + 3 + 3 + 3 + 3 = 15
Total Grade Points: 9 + 6 + 9.9 + 11.1 + 0 = 36
GPA = 36/15 = 2.4
```

5. Exit

The program should end only when option 5 is chosen. If the user enters an invalid option an error message should be printed and the menu re-displayed.

Files to Download

`students.txt:` student file

`courses.txt:` course file

`gradesheet.txt:` grades file

`COMP1602Assignment#2.cpp:` Use this file as a starting point for your solution to this assignment. Your code must be written in this `.cpp` file.

The file is to be renamed to your student id.

These four (4) files **must** be saved to a folder. The name of the folder is your student id number.

Note

Additional functions may be used.