**COSC 504**

**Spring 2020**

**Programming Project**

**Due May 5, 2020**

In this project, you are to implement a database. The database shows the students registered in different courses. The database will only include the student’s first name, last name, ID, list of courses, credit hours for each course, and the course grade. You are to calculate the GPA for each student. The GPA is calculated by multiplying the credit hours (which is given below for each course) by the grade value (A=4.0, B= 3.0, C=2.0, D=1.0, and F=0) for each course. Then you calculate the sum of the GPA for all the courses and divide it by the total credits of all the courses.

A sample of student’s information is listed below.

You will save all the information below in an input file called “SeniorClass”.

You will read from the file to perform the operations listed below.

**First Name Last Name ID Course Name Credits Grade**

James Smith 32145 Math411 3 A

James Smith 32145 Eng422 3 B

James Smith 32145 Phys355 4 A

James Smith 32145 Cosc361 4 B

Amanda George 31109 Math411 3 B

Chem344 4 B

Cosc361 4 B

Eng422 3 C

Lucy Williams 36749 Psyc388 3 C

Fren421 3 D

Hist410 3 C

Eng422 3 B

Anat390 3 B

Adam Howard 31546 Chem344 4 B

Math411 3 A

Hist410 3 C

Eng422 3 B

Phil477 3 C

George Brown 31220 Mus444 4 A

Art305 4 A

Eng422 3 B

Hist410 3 B

Willy Gates 30543 Eng419 4 A

Mrkt478 3 A

Econ366 3 B

Mrkt311 4 C

Ashley James 31321 Chem344 4 C

Math411 3 A

Hist410 3 B

Eng422 3 D

Phil477 3 C

Dan Edwards 34476 Math411 3 B

Eng422 3 C

Phys355 4 D

Cosc361 4 C

David Sammy 35109 Psyc388 3 D

Coun421 3 B

Hist410 3 C

Eng422 3 B

Anat390 3 B

Racheal John 38850 Mus444 4 B

Art305 4 B

Eng422 3 C

Hist410 3 A

Brandon Jackson 35561 Chem344 4 C

Math411 3 C

Hist410 3 C

Eng422 3 D

Phil477 3 B

Dina Lowe 36741 Psyc388 3 A

Coun421 3 D

Hist410 3 B

Eng422 3 D

Anat390 3 C

Brad Green 30179 Eng419 4 C

Mrkt478 3 B

Econ366 3 A

Mrkt311 4 B

Kevin Day 37409 Math411 3 A

Chem344 4 A

Cosc361 4 A

Eng422 3 A

Katty Hill 39137 Math411 3 B

Eng422 3 B

Phys355 4 C

Cosc361 4 B

Philips Bush 32210 Chem344 4 B

Math411 3 A

Hist410 3 A

Eng422 3 B

Phil477 3 A

Hellen Edwards 38889 Psyc388 3 C

Coun421 3 A

Hist410 3 A

Eng422 3 C

Anat390 3 D

Heidi Abraham 37701 Psyc388 3 B

Fren421 3 A

Hist410 3 B

Eng422 3 A

Anat390 3 A

Brandon Lowe 34402 Eng419 4 A

Mrkt478 3 C

Econ366 3 A

Mrkt311 4 A

Tiffany Green 36667 Chem344 4 A

Math411 3 A

Hist410 3 A

Eng422 3 A

Phil477 3 A

Edward Smith 36560 Chem344 4 A

Math411 3 A

Hist410 3 A

Eng422 3 A

Phil477 3 B

Display a Menu, so the user would be able to select any of the following operations:

CreateClassList

InsertNewStudent

AddCourse

DeleteStudent

DeleteCourse

FindGPA

Sort (Based on Last name, First name, GPA)

Search

Update

HonorStudents

WarnStudents

FailStudents

Print

Modify

Quit

Your program should include all the following functions:

**CreateClassList:** Reads the student’s information from the input file and place all student’s names along with their information in a linked list, in alphabetically order, by last name. Call the print function after you create the list.

**InsertNewStudent:** Will insert a new student into the class, along with all the student’s information as: first name, last name, ID, course name, credits, and grade, in the proper location in the list sorted alphabetically by last name. After insertion, you should show the count of the class. Then print the list after you insert.

**AddCourse:** You should be able to add a new course given the student’s name or ID, make sure to do all the necessary updates once you add a course.

**DeleteStudent:** You should be able to delete the student’s record, once you enter student’s first name followed by last name, or by using the student’s ID. If the student is not registered in the class, a message should appear indicating that the student is not registered. Show the count of the class after deleting. Then print the list after you delete. If any students have same first name and same last name, then you will have to delete by ID.

**DeleteCourse**: You should be able to delete any of the courses a student is registered in. To do so, you will have to ask for the student’s name or ID, and the course that needs to be deleted. Make sure to do all necessary updates after the deletion.

**FindGPA:** Calculates the GPA for any student by multiplying the credit hours (which is given below for each course) by the grade value (A=4.0, B= 3.0, C=2.0, D=1.0, and F=0) for each course. Then you calculate the sum of the GPA for all the courses and divide it by the total credits of all the courses.

**Sort:** You should be able to sort the list by first name, GPA, and ID. You could use any sorting algorithm. Then print the list after you sort. You should ask the user how you want to sort, either by first name, or GPA, or ID. When you print, you will have to use a different print function than the one described below, because the one below prints the names sorted alphabetically by last name.

**Search:** By giving the student ID, you should be able to search for that student in the list. You should return the student’s first name, last name, ID, GPA, and the courses registered. If the student is not found in the list, you should print a message indicating so.

**Modify:** You should be able to modify the student’s grade . So you should be able to ask the user for the student’s name or ID, then ask what course grade needs to be modified. You should show update the student’s information and print the updated student’s information.

**HonorStudent:** Prints the number of students and names of students whose GPA is greater than 3.6.

**WarnStudent:** Prints the number of students and the names of the students whose GPA is less than 2.5.

**FailStudent:** Prints the names of students whose GPA is less than 2.0

**Print:** Prints all the students information alphabetically, sorted by last name, and the total number of students in the class.