**Data Structures- Fall 2022**

Grade Report

The mid-semester point at your local university is approaching. The register's office wants to prepare, the grade reports as soon as the student’s grades are recorded. However, some of the students enrolled have not yet paid their tuition.

1. If a student has paid the tuition, the grades are shown on the grade report together with the grade point average (GPA).
2. If a student has not paid the tuition, the grades are not printed. For these students, the grade report contains a message indicating that the grades have been held for nonpayment of the tuition. The grade report also shows the billing amount.

The Registrar’s Office and the Business Office want your help in writing a program that can analyze the students’ data and print the appropriate grade reports. The data is stored in a file in the following form:

**1500 345**

**studentName studentID isTuitionPaid numberOfCourses courseName courseNumber creditHours grade courseName courseNumber creditHours grade**

**.**

**.**

**.**

**studentName studentID isTuitionPaid numberOfCourses courseName courseNumber creditHours grade**

**courseName courseNumber creditHours grade**

**.**

**.**

**.**

The first line indicates the number of students enrolled and the tuition rate per credit hour. The students’ data is given thereafter.

A sample input file is as follows:

**3 345**

**Lisa Miller 890238 Y 4 Mathematics MTH345 4 A Physics PHY357 3 B ComputerSci CSC478 3 B History HIS356 3 A**

**.**

**.**

**.**

The first line indicates that the input file contains three students’ data, and the tuition rate is $345 per credit hour. Next, the course data for student Lisa Miller is given. Lisa Miller's ID is 890238, she has paid the tuition and she is taking four courses. The course number for the mathematics class she is taking is MTH345, the course has four credit hours, her mid-semester grade is A, and so on.

The desired output for each student is in the following form:

Student Name: Lisa Miller Student ID: 890238

Number of courses enrolled: 4

|  |  |  |  |
| --- | --- | --- | --- |
| **Course No** | **Course Name** | **Credits** | **Grade** |
| CSC478 | ComputerSci | 3 | B |
| HIS356 | History | 3 | A |
| MTH345 | Mathematics | 4 | A |
| PHY357 | Physics | 3 | B |

Total number of credits: 13 Mid-Semester GPA: 3.54

It is clear from this output that the courses must be ordered according to the course name. To calculate the GPA, we assume that the grade A is equivalent to four points. B is equivalent to three points. C is equivalent to two points. D is equivalent to one point, and F is equivalent to zero points.

**Input** A file containing the data in the form given previously is provided for you in Convas. For every reference, let us assume that the name of the input file is **stData.txt**.

**Output** A file containing the output in the form given previously.

We must first identify the main components of the program. The university has students, and every student takes courses. Thus, the two main components are the students and the course.

Let us first describe the course and the student component.

**Course:**

The main characteristics, of course, are the course name, course number, and number of credit hours.

Some of the basic operations that need to be performed on the course structure are:

1. Set the course information
2. Print the course information
3. Show the credit hours
4. Show the course number

**Student**

The main characteristics of a student are the student’s name, student ID, number of courses in which enrolled, courses in which enrolled, and grade for each course.

Because every student has to pay tuition, we also include a member to indicate whether the student has paid the tuition.

The basic operations to be performed on a student structure are as follows:

* 1. Set the student information.
  2. Print The student information.
  3. Calculate the number of credit hours taken
  4. Calculates GPA.
  5. Calculate the billing amount.
  6. Because the grade report will print the courses in ascending order, sort the courses according to the course number.

**Main Program**

In your main program your will use a ‘Double Linked List’. It does not have to be a ‘Circular’ one.

The program should read from the file “**stData.txt**” text file provided for you**.** It should start with the following menu:

* + 1. **Add a student**
    2. **Update a student’s course Grade**
    3. **Delete a student**
    4. **Search for a student**
    5. **Exit**
* When choosing 1, the program must ask the user to insert the id of the student, followed by all other information for the student (Name, number of courses enrolled, paid the tuition or not, the name of each course, the number of each course, the number of credits of each course taken, the letter grade obtained for each course. If entering an id that already exists, you must display a message saying that the student already exists, and the user must try entering a different id.
* When choosing 2, the program must ask the user to enter the id of the student to be updated. If the id entered does not exist in the file, an error message should be displayed. If the id exists, the program must ask the user which course number he/she wants to update its grade, then the user must enter the new grade. The program, then should ask the user if he/she wants to update another course grade. If not, the program must display the menu.
* When choosing 3, the program must ask the user to enter the id of the student to be deleted. If it does not exist, the program must show an error message. Otherwise, the student must be removed from the file.
* When choosing 4, the program must ask the user to enter the id of the student looking for. If it does not exist, the program must show an error message. Otherwise, there are two cases:
  + if the student did not pay the tuition, A message must be displayed on the screen saying that the grade report cannot be displayed until the tuition is paid, but his/her information must be displayed on the output file, in the form given previously in red.
  + If the tuition was paid, the information of that student should be displayed, on the screen and on the output file, in the form given previously in red.
* When choosing 5, the input file must be saved, and the program must be exited.
* **Note** that every time, you enter a new student, update a student, or delete a student, the new data must be saved in the input file “**stData.txt**”.