**Use your time wisely! No late submission will be taken!**

**Please note, if your codes are not compiling or cannot be executed, they will be graded as 0.**

Write a Windows Forms application follow the instructions below.

1. **(5 points)**Define an **interface** **Register**, it has the methods:

**Eligibility** (): will determine whether this course can be registered or not, i.e., **available seat** is bigger than 1 and the **pre-requisites** meet. Please note, it might need a parameter of Course class defined below.

**UpdateSeat()** which will update the number of seats available.

All methods will be declared here and implemented in the Course class.

**Please note you need to determine the parameter (s) and returned type of those methods. You can add any methods to make your work easier**.

1. **(15 points)** Please define a class **Course** implement the previously defined **Register** interface. It has **name**, **available seat**, **pre-requisites** (there might be upto **two** pre-requested courses) attributes (you can determine the identifiers and types for those attributes).

**Please note you need to implement the methods declared in Register interface and can add any attributes/methods to make your work easier**.

1. **(10 points)** Define an abstract **class Student**, which has **student ID**, **name**, **course taken** (the courses have been taken, let’s assume all students have taken upto 3 courses, **it is an array of Course class**), **course to register** (we assume there is just one course to register), **course register status** ( true or false to indicate the course is registered successfully or not).

Declare methods within Student class:

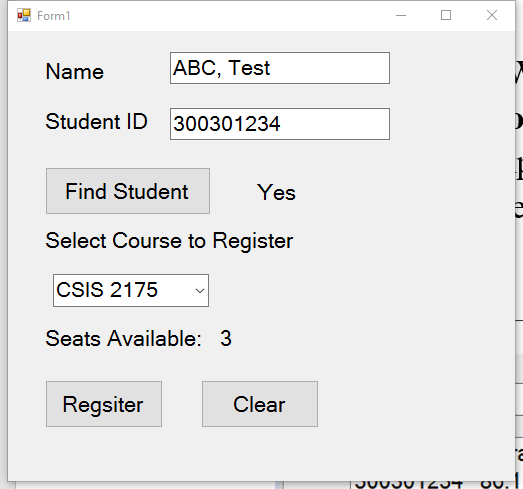
**CourseRegistration**(): will register the course by setting the course register status as True.

**Please note that you need to determine the return type and parameter list by yourself, but keep the names as stated.** **You can add any attributes/methods to make your work easier**.

1. **(10 points)** Declare class **CollegeStudent** which extend **Student**. It has one more attribute **tuition**, where if the student is able to register the course, it will be $2000 otherwise it will be $100 (let’s say it is auditing).

It will then override the **CourseRegistration()**  method: If the course is fully registered or student doesn’t meet the pre-requisite, the course register status will be False -- NEED TO IMPLEMENT YET; if student is able to register, it will be true then. Please note, if the course has been taken by the student, he is able to register with half of the standard tuition, i.e, $1000. The available seat (defined in Course) will be deducted by 1 if the student is able to register and remains the same if not.

1. (**10 marks**)Display a menu (Windows Forms based, NOT the console). There will also be button, label, textbox and combo-box etc. (or other components you select). The sample result is shown below, your layout and design can be different, but all the components must be there.



Please note the combobox displayed above has 3 items: CSIS1175, CSIS2175 and CSIS2300.

1. (**20 marks**)Using the class and GUI to implement the course registration procedure :
   1. User will input name and student ID and then click Find Student button (as in the GUI above), if the student’s name and ID are found in the file **student.txt**, display Yes, otherwise display No and clear all input. Hint: you need to read the file.

Besides, an object of **CollegeStudent**, will be created using the student ID, name and other information of that student. The format is :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Student ID | Name | Course 1 taken | Course 2 taken | Course 3 taken |

* 1. Then user will select the course from the combobox (as in the GUI above), then the number of seats available will be displayed after the Seats Available label. The information is contained in the **course.txt. The format is :**

|  |  |  |  |
| --- | --- | --- | --- |
| Course Name | Available seats | Pre-requisite 1 | Pre-requisite 2 |

**Please note if the pre-requisite can be “none”, which means the course might have 0 or just 1 pre-requisite.**

* 1. Then user click register button, the **CourseRegistration()** method will be invoked (the course to be registered is obtained from the combobox), the available seat and course register status attributes will be updated.
  2. The register result will be written to a file, **record.txt**, the format is

|  |  |  |  |
| --- | --- | --- | --- |
| Student ID | Student Name | Course | Tuition |

**Please note if the student is not able to register the course, still write the information to the file.**

**Please note that you can determine any unmentioned details.**