Assignment 2

Please read the description for each classes below and implement these classes.

You may need to add new functions or attributes that are not mentioned in the description of specific classes. So add any functions or attributes you need in any class to implement the requirements

Notes:

1. **Individual assignment**
2. **Submission on moodle**
3. **File Name is NAME 1- ID 1.rar**
4. **Due date 25/12/2015**
5. **Cheaters will be graded by –ve points**

Class Student

Attributes

- string name
- int level
- Course *courses (Array of courses)
- int totalNumberOfCourses;

Methods

- Default constructor
- Parameterized constructor that accepts name, level and total number of courses
- Copy constructor
- Method to add new course to the courses array. This function should accept a course as a parameter and it should add this course to courses array
- Method to calculator the total credit hours for a student. The implementation of this function will depend on student type so this function should be pure virtual. This method should return double data type
- Destructor
- Setter and getter (for name and level attribute)
- Overload operator=

Class GraduateStudent

This class should inherit from student class (public inheritance)

Methods

- Default constructor
- Parameterized constructor, the same parameterized constructor used in student class so this constructor should call student constructor
- Method to calculate the total credit hours for graduate students. The credit hours for a graduate student equals to the summation of the credit hour for each course. Graduate students take 2 labs and 2 lectures for each course. So to calculate the credit hour of specific course we multiply lab time by 2 and lecture time by 2

  So if there's a course which lab time is 1 hour and lecture time is 1.5 and section time is 1.5. So the credit hour for this course is 1*2 + 1.5*2 = 1 + 3 = 4 hours

- Destructor

Class NormalStudent

This class should inherit from student class (public inheritance)

Methods

- Default constructor
- Parameterized constructor, the same parameterized constructor used in student class so this constructor should call student constructor
- Method to calculate the total credit hours for graduate students. The credit hours for a normal student equals to the summation of the credit hour for each course. Normal students take 1 lab and 2 lectures and 1 section for each course. So to calculate the credit hour of specific course we multiply lab time by 1, lecture time by 2 and section time by 1

  So if there's a course which lab time is 1 hour and lecture time is 1.5 and section time is 1.5. So the credit hour for this course is 1*1 + 1.5*2 + 1.5*1 = 1 + 3 + 1.5 = 5.5 hours

- Destructor

Class Course

Attributes

- string name
- double lectureTime
- double labTime
- double sectionTime

Methods

- Default constructor
- Parameterized constructor (name, lectureTime, labTime, sectionTime)
- Destructor
- Setter and getter for each attribute

Class SemesterManagement

This class will manage the process of add new student to the system. Also this class is responsible for showing students information (like the total already spent credit hours for each student)

Attributes:

- Array of students (Hint, this array should be array of students' pointers)
- Total number of students (the size of the students array)

Methods:

- Default constructor
- Parameterized constructor, takes the size of students array and initialize students array with the given size
- A method that takes a student pointer and add this student to the array of students
- A method to print a report for each student in the students' array. For each student you should print his/her name and the total number of his/her credit hours
- Destructor

Example input

```
NormalStudent s1("Mohamed" , 1 , 6);
/*
*  Suppose csCourse, englishCourse, programmingCourse and
management course are courses object already defined before
* Also assume that the lab time, section time and lecture time
for all these courses are the same (1.5 hours) except that the
section time for English course is 3 hours
*/
s1.addCourse(csCourse);
s1.addCourse(englishCourse);
s1.addCourse(managementCourse);

NormalStudent s2;
s2 = s1;
s2.setName("Ahmed");
s2.setLevel(2);
s2.addCourse(programmingCourse);

GraduateStudent s3("Ibrahim" , 1 , 6);
s3.addCourse(csCourse);
s3.addCourse(englishCourse);
s3.addCourse(managementCourse);

SemesterManagement sm(2);
sm.addNewStudent(&s1);
sm.addNewStudent(&s2);
sm.addNewStudent(&s3);
sm.printStudentsReport();
```

The output should be

```
Student name: Mohamed
Current hours are: 19.5
Student name: Ahmed
Current hours are: 25.5
Student name: Ibrahim
Current hours are: 18
```