Project 3

Course Management System

CSCI 111 Programming and Algorithms I

Due Friday November 22, 11:59PM on ***turnin***

60 Points

NEW CONCEPTS: Successful completion of this lab incorporates the following new concepts.

- None

Create a **Course Management System** that allows the manage course object(s) and student object(s) by creating, storing, and displaying their contents. Completion of this Project 3 is critical since Project 4 extends the capability with removal of course/student objects, plus more. Provide the user with the following menu:

1) Add a new course
2) Add a new student
3) Display all the courses
4) Display all the students within a course
5) Remove a course (Project 4 will implement this feature)
6) Remove a student within a course (Project 4 will implement this feature)
7) Exit

Your project should consist of the following 4 files: **project3.cpp**, **activity.cpp**, **activity.h**, and the **Makefile**. The code in project3.ccp should be minimal and only initialize the root level data structure pointer. Most of the code should be written in the **activity.h** file. And, **activity.h** should contain and use the following subset code:

```
class Student
{
    private:
        int studentId;
        string studentName;
    public:
        Student(int i1, string s1)
        {
            studentId = i1;
            studentName = s1;
        }
        string getName()
        {
            return studentName;
        }
};

class Course
{
    private:
        string courseName;
        int numUnits;
        vector<Student> vs;
    public:
        Course(string s1, int i1)
        {
            courseName = s1;
            numUnits = i1;
        }
        int getUnits()
        {
            return numUnits;
        }
};
```

Task list:

1. Create a source code file named **project3.cpp, activity.cpp** and **activity.cpp**.

2. The program starts by prompting the user with the menu listed above

3. Add a new course: Prompt the user for a *courseName* and *numUnits*. Create an object of the course and add it to the system. The ***new*** operator and a class ***constructor*** should be used to set the values. Prior to creating a new vector object, ensure that the *courseName* has not be added and issue an error message accordingly.

4. Add a new student: Prompt the user for a *courseName*, *studentId*, and *studentName*. Create an object of the student and add it to the respective *courseName*. The ***new*** operator and a class ***constructor*** should be used to set the values. Prior to adding a new vector object, ensure that the *studentId* has not be added and issue an error message accordingly.

5. Display all the courses: Traverse the course objects loaded to display all the *courseName* values using a 'getter' function.

6. Display all the students within a course: Prompt the user for a *courseName* and then traverse the student courses within the given *courseName* to display all the *studentId* values using a 'getter' function.

7. Do not implement the dot notation to access object contents. Use class member functions.

8. Submit your **activity.cpp** and **activity.h** file on https://turnin.ecst.csuchico.edu/.

GRADING

To achieve a maximum score, students will need to clearly prove that they completed the goal.

- Points lost for incompleteness, sloppiness, lateness, or failure to follow instructions.
- Refer to syllabus for late policy.

SAMPLE OUTPUT

```
jraigoza@ecc-jaguar:~/csci111/project3$ ./project3
1) Add a new course
2) Add a new student
3) Display all the courses
4) Display all the students within a course
5) Remove a course (Project 4 will implement this feature)
6) Remove a student within a course (Project 4 will implement this feature)
7) Exit
1
Enter the new course name.
CSCI 111
Enter the number of units.
4
Adding course
1) Add a new course
2) Add a new student
3) Display all the courses
4) Display all the students within a course
5) Remove a course (Project 4 will implement this feature)
6) Remove a student within a course (Project 4 will implement this feature)
7) Exit
```

```
1
Enter the new course name.
MATH 120
Enter the number of units.
4
Adding course
1) Add a new course
2) Add a new student
3) Display all the courses
4) Display all the students within a course
5) Remove a course (Project 4 will implement this feature)
6) Remove a student within a course (Project 4 will implement this feature)
7) Exit
3
CSCI 111
MATH 120
1) Add a new course
2) Add a new student
3) Display all the courses
4) Display all the students within a course
5) Remove a course (Project 4 will implement this feature)
6) Remove a student within a course (Project 4 will implement this feature)
7) Exit
2
Enter the course name.
CSCI 111
Enter the students ID #.
123
Enter the students name.
Jose
Adding student
1) Add a new course
2) Add a new student
3) Display all the courses
4) Display all the students within a course
5) Remove a course (Project 4 will implement this feature)
6) Remove a student within a course (Project 4 will implement this feature)
7) Exit
2
Enter the course name.
MATH 120
Enter the students ID #.
456
Enter the students name.
Ruben
Adding student
1) Add a new course
2) Add a new student
3) Display all the courses
4) Display all the students within a course
5) Remove a course (Project 4 will implement this feature)
6) Remove a student within a course (Project 4 will implement this feature)
7) Exit
2
Enter the course name.
CSCI 111
Enter the students ID #.
```

```
789
Enter the students name.
Sandy
Adding student
1) Add a new course
2) Add a new student
3) Display all the courses
4) Display all the students within a course
5) Remove a course (Project 4 will implement this feature)
6) Remove a student within a course (Project 4 will implement this feature)
7) Exit
1
Enter the new course name.
PHYS 204
Enter the number of units.
4
Adding course
1) Add a new course
2) Add a new student
3) Display all the courses
4) Display all the students within a course
5) Remove a course (Project 4 will implement this feature)
6) Remove a student within a course (Project 4 will implement this feature)
7) Exit
3
CSCI 111
MATH 120
PHYS 204
1) Add a new course
2) Add a new student
3) Display all the courses
4) Display all the students within a course
5) Remove a course (Project 4 will implement this feature)
6) Remove a student within a course (Project 4 will implement this feature)
7) Exit
4
Enter the course name.
CSCI 111
Jose
Sandy
1) Add a new course
2) Add a new student
3) Display all the courses
4) Display all the students within a course
5) Remove a course (Project 4 will implement this feature)
6) Remove a student within a course (Project 4 will implement this feature)
7) Exit
4
Enter the course name.
MATH 120
Ruben
1) Add a new course
2) Add a new student
3) Display all the courses
4) Display all the students within a course
5) Remove a course (Project 4 will implement this feature)
6) Remove a student within a course (Project 4 will implement this feature)
```

```
7) Exit
4
Enter the course name.
PHYS 204
1) Add a new course
2) Add a new student
3) Display all the courses
4) Display all the students within a course
5) Remove a course (Project 4 will implement this feature)
6) Remove a student within a course (Project 4 will implement this feature)
7) Exit
7
jraigoza@ecc-jaguar:~/csci111/project3$
```