EECS484 Database Management Systems
Homework 2 (90 points)
Due 11:55PM, February 4th 2016

Please read the following instructions before starting the homework:

You should have access to the following files:
booktown.sql
booktownqueries_problems.sql
drop_booktown.sql

There are two parts to this homework:

The **first** part requires you to write SQL queries given a schema. We will not be providing you with a database for this question. It is advised you think of some way to be able to check if your queries really work they way they should. Please submit this part in a file called hw2part1.sql.

The **second** part requires you to look at the database associated with a sql file called *booktown.sql*. This file should be attached along with this document on canvas. You will need your Oracle account to load and see the database. Please answer question 4 in the file named *booktownqueries_problems.sql*. Question 5 is to be submitted through canvas in a pdf file. Please neatly write the relational algebra and relational calculus questions and scan them with a high quality scanner (either a good phone or a scanner found in the library). Alternatively, you can write them using an equation writing software (e.g. LaTeX).

You are *allowed* to work on this homework in pairs.

If you are working in a pair:
Please join the same group with your partner on Canvas (refer to the announcement for more information on joining groups). We will be expecting only one submission per group.

No late days for homework! If you miss the due date, you get 0 points. No exceptions on this.

By submitting this homework, you are agreeing to abide by the Honor Code:
I have neither given nor received unauthorized aid on this assignment, nor have I concealed any violations of the Honor Code.

Part 1 (30 points)

Write some queries for the given relational schema. You are given the following relations (primary keys are underlined):

Student (**sid**, name, major) : student id, name, and majors of students
Project (**pid**, ptitle) : project ID and title of projects
Course (**cid**, title) : course ID and title of courses
Member (**pid**, **sid**) : Relationship. Student sid is a member of project pid
Enrolled (**sid**, **cid**) : Relationship. Student sid is enrolled in course cid.

Please submit the answers to these questions in a file called **hw2part1.sql**

Question 1 (10 points):
Find the name and student id of all students who have a project partner who is enrolled in **'EECS484'** AND either of '**EECS483**' OR '**EECS482**' (these are the titles of a course). For example, student A will be the output if students A and B are members of the same project and B is enrolled in (EECS484 && (EECS483 || EECS482))
Correct solutions that use views, nested queries, intersections, or unions will receive only 50% credit. (This query can be done without views, nested queries, intersections, or unions)

Question 2 (10 points):
We would like to know the non-CS majors who are taking heavily CS-based courses. We define heavily CS-based courses as courses where more than 60 'CS' majors are enrolled. Return all the student id (sid) and names of non-CS students, as well as the class title, where more 60 'CS' majors are enrolled in that class. The result of the query should be output in decreasing order by name.
You may use views or nested queries for this part.

Question 3 (10 points):
The school is interested in seeing which students are actively working together in projects that are not course-related. Define a VIEW that contains all student id pairs (sid1, sid2) with the following property: the students who are members of the same project, but are not enrolled in a common course. List each pair of students only once. For example, if you list (23, 42), do not list (42, 23). Student ids can be assumed to be integers.
You may use views or nested queries for this part.
Hint: This one is actually quite subtle and tricky! Try not to lose potential student pairs when joining your tables!

Part 2 (60 points)

We have provided you a sample database in **booktown.sql**. Please look through the beginning of this file to understand the schema. Feel free to add more sample data to test your answers. However, don't change the schema! We *will* be testing your answers using the schema that we have given you.

To build the database in SQLPlus:
Log into CAEN and navigate to the directory where booktown.sql is located at. And run the following:

```
$ module load oracle
$ sqlplus
<type username and password>
SQL> START booktown.sql
```

Question 4 (40 points):
- Please provide queries to the questions listed in **booktownqueries_problems.sql**.
- There should be 8 questions in total in this file.
- Your queries should work for any database with the schema provided in **booktown.sql**, *not* just with the same data we have given you.
- If you ever make intermediate view, make sure you drop them at the end of that question.

IMPORTANT: Ensure that once you are done, the entire **booktownqueries_problems.sql** can run completely without errors by:
```
SQL> START booktownqueries_problems.sql
```

Question 5 (20 points):
Write relational algebra *and* relational calculus expressions for the queries Q1, Q2, Q3 and Q4 that you wrote in the **booktownqueries_problems.sql** file above. These do not involve aggregation and should be doable with relational algebra.

Submission

Ensure that you submit the following files onto canvas. Any missing files will not be accepted after the deadline, and you will be given a 0 for it.

Files to be submitted to canvas:
- ❏ **hw2part1.sql**
- ❏ **booktownqueries_problems.sql**
- ❏ A .pdf file containing RA and RC expressions for Part 2 Question 2

Make sure your .sql files can run in a CAEN environment using SQLplus from start to finish. Failure to run (in any way) will result in you getting a 0. MAKE SURE IT WILL RUN!!