Comp115 Spring 2017, HW3 - Due March Xth, 2017

Submit via provide using:

```
provide comp115 115HW3-RA-SQL <single_file.pdf>
```

Problem 1 - 25%

Consider the following schema:

> Publishers(*pid:* `integer`, *pname:*`string`, *address:* `string`)
> Books(*bid:* `integer`, *bname:*`string`, *length:* `integer`)
> TuftsBookstore(*pid:* `integer`, *bid:*`integer`, *cost:* `real`)

The key fields are underlined and the domain of each field is listed after the field name. Therefore *bid* is the key for Books, *pid* is the key for Publishers, and *bid* and *pid* together form the key for TuftsBookstore. The TuftsBookstore relationship lists the prices charged for textbooks by Publishers. Write the following queries in relational algebra:

1. Find the *names* of publishers who supply books over 1000 pages long.
2. Find the *pids* of publishers who supply books under 200 or over 1000 pages long.
3. Find the *pids* of publishers who supply books over 1000 pages long and are at 161 College Ave.
4. Find the *pids* of publishers who supply books between 200 and 1000 pages long.
5. Find the *pids* of publishers who supply every book.
6. Find the *pids* of publishers who supply every book over 1000 pages long.
7. Find the *pids* of publishers who supply every book under 200 or over 1000 pages long.
8. Find the *pids* of publishers who supply every book under 200 pages long or who supply every book over 1000 pages long.
9. Find pairs of *pids* such that the publisher with the first *pid* charges more for some book than the publisher with the second *pid*.
10. Find the *bids* of books that are supplied by at least two different publishers.
11. Find the *bids* of the most expensive books published by Pearsons.
12. Find the *bids* of books supplied by every publisher for less than $200. (If any publisher either does not supply the book or charges more than $200 for it, the book is not selected).

Problem 2 - 25%

Please state in lay terms what the following queries compute

1. $\pi_{pname}\left(\pi_{bid}\left(\sigma_{length>200}Books\right) \bowtie \left(\sigma_{cost<100}TuftsBookstore\right) \bowtie Publishers\right)$
2. $\pi_{pname}\left(\pi_{pid}\left(\left(\sigma_{length>200}Books\right) \bowtie \left(\sigma_{cost<100}TuftsBookstore\right)\right) \bowtie Publishers\right)$
3. $\left(\pi_{pname}\left(\left(\sigma_{length>200}Books\right) \bowtie \left(\sigma_{cost<100}TuftsBookstore\right) \bowtie Publishers\right)\right) \cap$
   $\left(\pi_{pname}\left(\left(\sigma_{length<300}Books\right) \bowtie \left(\sigma_{cost<100}TuftsBookstore\right) \bowtie Publishers\right)\right)$
4. $\left(\pi_{pid}\left(\left(\sigma_{length>200}Books\right) \bowtie \left(\sigma_{cost<100}TuftsBookstore\right) \bowtie Publishers\right)\right) \cap$
   $\left(\pi_{pid}\left(\left(\sigma_{length<300}Books\right) \bowtie \left(\sigma_{cost<100}TuftsBookstore\right) \bowtie Publishers\right)\right)$
5. $\pi_{pname}(\left(\pi_{pid,pname}\left(\left(\sigma_{length>200}Books\right) \bowtie \left(\sigma_{cost<100}TuftsBookstore\right) \bowtie Publishers\right)\right) \cap$
   $\left(\pi_{pid,pname}\left(\left(\sigma_{length<300}Books\right) \bowtie \left(\sigma_{cost<100}TuftsBookstore\right) \bowtie Publishers\right)\right))$

Problem 3 - 50%

Consider the following relations:

Student(*snum:* `integer`, *sname:*`string`, *major:*`string`, *level:*`integer`, *age:*`integer`)
Class(*name:*`string`, *time:*,`time`, *room:*,`string`, *fid:*`integer`)
Enrolled(*snum:*,`integer`, *cname:*`string`)
Faculty(*fid:*`integer`, *fname:*`string`, *deptid:*`integer`)

The meaning of these relations is straightforward; for example, Enrolled has one record per student-class pair such that the student is enrolled in the class. The level is 1 for Freshman, 2 for Sophomore, 3 for Junior, and 4 for Senior. Write the following queries in SQL. No duplicates should be printed for any answer.

1. Find the names of all students who are enrolled in 0 classes.
2. Find the level of all students who are enrolled in a class that starts before 9:00AM.
3. Find the course with the most students enrolled that starts before 9:00AM.
4. Find the number of unique students that every professor teaches.
5. Find the names of all Sophomore ($level = 2$) Computer Science majors who are enrolled in a class taught by Mark Sheldon.
6. Find the name of the youngest student who is an American Studies major or in an Intro to International Relations class.
7. Print the average level of students in each class, for every class.
8. Print the average age and average level of students in each major, for every major.

9. Find the major in which the most students have more than one class with a given professor.
10. Find all pairs of students taking the same courses.
11. Find all pairs of students taking courses from the same professors.
12. For each major, find the student with the largest gap in their schedule (time between two classes).