CS127 Homework #1

Due: September 18, 2019 6:00 P.M.

Handing In

Upload your homework to gradescope.
Please write your Banner ID on your submission. Do not write your name on the submission.

Grading Information

Grading for the homeworks works as follows:

- The set of warm-up problems will be graded as one of (✓+, ✓, ✓-)
- All other problems will be graded in detail and will be given a score.

Solutions for the warmup problems will be provided along with your graded work.

Warmup #1

Consider the following database schema for the dealerships around the world:

- *dealership(dealership_name, city, country)*
- *car(car_id, car_name, dealership_name, year_made)*

Given that:

- Every dealership has a unique name.
- Any two cars may have the same name.
- The cars in different dealerships may have the same id, but not the cars in the same dealership.

Identify all potential super keys, candidate keys, and primary keys for the schemas above.

Warmup #2

Given the following database schema:
driver(driver_id, car_id, driver_name, age, state)
car(car_id, model, year_produced)
Write a relational algebra expression for the following questions:

1. Find the ids of drivers older than 45 living in Rhode Island.
2. Find the names of the drivers driving a car produced between the years 1990 and 2000.
3. Find the car ids not registered with drivers who live in Rhode Island.
4. Find the average year produced of each model of car.

Graded Problems

For the following graded problems, use the schema below:

department(dept_name, building, budget)
course(course_id, title, dept_name, credits)
instructor(i_ID, name, dept_name, salary)
section(course_id, sec_id, semester, year, building, room_number, time_slot)
teaches(i_ID, course_id, sec_id, semester, year)
student(s_ID, name, dept_name, tot_cred)
takes(s_ID, course_id, sec_id, semester, year, grade)
prereq(course_id, prereq_id)

Problem 3

Write the following queries in relational algebra using the schema above (10 points each)

1. Find the ID and name of each student who has *not* taken a course from a department in the building "CIT" (10 points)
2. Find the course and section IDs for sections with conflicting time slots (10 points)
3. Find the ID and name of each instructors who teach courses in the "Computer Science" or "Math" department or teaches a course with no prerequisites (10 points)

Problem 4

G_{max} can be expressed in relational algebra without aggregation. We'll build toward this relational algebra below to answer the query:

> Find the department name with the most number of students taking at least one of the department's classes. There may be more than one department with the most number of students.

1. Write the relational algebra to find department name and the number of students taking at least one of its classes. (Hint: you should use G_{count} here). Rename the resulting attribute containing the number of students as *count*. Let's call this relation α. (10 points)
2. Let's find the tuples in α that don't qualify. That is, for a tuple t in α, there exists another tuple s where $t_{count} < s_{count}$.
   (a) Write the relational algebra that results in a relational that is the cartesian product of α *with itself*. Remember to prevent ambiguous attributes! (2 points). The resulting relation should have 4 attributes: 2 from each side of the product.
   (b) The tuples we don't want are where the count from one side is $<$ the count from the other side of the cartesian product. Can you see why? Write the relational algebra to find these tuples from your result from 2.1 above. (2 points)
   (c) Write the relational algebra that removes the tuples from 2.2 above from α. (2 points)
3. Combine the steps from 2 above to write the relational algebra to answer our original query. In other words, find the tuple in α with the maximum *count*. For brevity, you can ignore the *attribute renamings* (but not the relation renamings) you wrote above. You shouldn't need the assignment operator for this one-liner(4 points)

Problem 5

The division ($\div$) relational operator identifies attribute values from a relation that can be paired with all of the values from the other relation. One way to think about it is that it is the inverse of the Cartesian Product ($\times$) relational operator: given relations R and S, $(R \times S) \div R = S$.

In the following questions, we'll derive the relational operators that make this operation possible by answering the query:

> Find the IDs of all students who have taken all "Math" courses.

As you are going through these, make sure you understand what attributes you are projecting during each step. Here we go!

1. It seems we'll need at least two relations. One for all students and courses, and one for all "Math" courses. Write the relational algebra to answer the following queries (10 points each):
   (a) Find all student IDs and course IDs for all "Math" courses. Assign this result to α. (Hint: this should be a relation with s_ID and $course\_id$)
   (b) Find the course IDs for all "Math" courses. Assign this result to β. (Hint: this should be a relation with just $course\_id$)
   (c) The list of possible answers are in $\alpha(s\_ID)$. Write the relational algebra to find all possible s_ID and $course\_id =$ "$Math$" pairs. Let's call this γ.
2. The idea is to find, in γ, all pairings that *do not exist* and then remove these from the possible answers. You'll be left with the right answers! (5 points each).
   (a) The pairings in γ that actually exist can be found in α. Write the relational algebra to find all pairings in γ that *do not exist*. Let's call this δ. This is the key key step in the process! Can you explain to yourself why this is?
   (b) Finally, we can remove the non answers in δ. Write the relational algebra to find all possible answers s_ID that are not in δ. (This is a confusing double negative: find all possible answers that are not in the list of impossible answers)
3. Putting it all together, write the relational algebra that defines $\alpha \div \beta$ where $\alpha(s\_ID, course\_id)$ and $\beta(course\_id)$. Note that it's important to understand what we are projecting in each of the steps in the relational algebra. Also describe in 3 steps what this algebra is doing. (10 points)