Database Description

The entirety of this exam will work with an example database storing information for an online Pacific NW animal shelter management system. You will use this database to write relational algebra expressions as well as SQL DDL and DML.

Scenario: You are tasked with managing a Pacific NW animal shelter online management system, which manages information about a small collection of shelters, animals, employees, applications, and applicants for adoption of various animals in Washington, California, and Oregon. A subset of data is provided for you in **`shelter-data.sql`**.

Here is the database schema that will be used for the online animal shelter system:

shelters(shelter_id, name, address, zipcode, city, state)
employees(emp_id, name, gender, is_volunteer, phone, email, role, join_date, shelter_id)
animals(animal_id, name, gender, animal_type, breed, age_est, notes, shelter_id, join_date, adoption_price)
applicants(applicant_id, name, phone, address, zipcode, curr_pet_count, household_size, notes)
applications(app_id, applicant_id, animal_id, status, application_date)

Relational Algebra *(35 points)*

1. You Can't Put a Price on Pets... But Shelters Don't Run Themselves. Write a query to retrieve the average adoption price for each animal type registered in the database.

$$_{animaltype}G_{AVG(adoptionprice)\ AS\ avgadoptionprice}(animals)$$

2. They Dream of Doggos. Write a query to retrieve the name, gender, estimated age, and breed of all dogs with applications having a status of "Submitted".

$$\Pi_{name,\ gender,\ ageest,\ breed}(\sigma_{animaltype\ =\ 'Dog'\ \wedge\ status\ =\ "Submitted"}(animals \bowtie applications))$$

3. The Wait Begins. Write a query that returns the name and phone number of all applicants who have an application with status "Submitted", as well as the animal id of different animals they have applied to adopt.

$$\Pi_{name,\ phone,\ animalid}(\sigma_{status\ =\ "Submitted"}(applicants \bowtie applications))$$

4. New Additions to Small Homes. Retrieve the name and animal type of all animals who *only* have applications from applicants who have 3 or fewer individuals living in the household.

$$animalinfo \leftarrow_{animalid}G_{MAX(householdsize)\ AS\ maxsize}(applications \bowtie \Pi_{applicantid,\ householdsize}(applicants))$$

$$\Pi_{name,\ animaltype}(animals \bowtie \Pi_{animalid}(\sigma_{maxsize\ <\ 4}(animalinfo)))$$

5. And You Said It Was Only Volunteering... Write a query which returns the name of all employees who have applied to adopt an animal at the shelter, as well as the id, name, and animal type of each animal they have applied for. Note that employee ids and applicants ids have different

meanings - assume that the name and phone number uniquely identifies an individual, so you'll want to look for records that match on these values between *employees* and *applicants*.

$$apps \leftarrow employees \bowtie applicants \bowtie applications$$

$$\Pi_{apps.name\ AS\ employeename,\ apps.animalid\ AS\ animalid,\ animals.name\ AS\ animalname,\ animaltype}(\sigma_{apps.animalid = animals.animalid}(apps \times animals))$$

6. Spam-Filtering. Let's make sure that we don't have any applicants that have spammed applications for every shelter in the database. Write a query which returns the name and phone-number of any applicant who has an application <u>for every</u> shelter currently in the *shelters* relation. *Hint: What extended relational algebra operation can be useful here?*

$$shelterinfo \leftarrow \Pi_{shelterid}(shelters)$$

$$appinfo \leftarrow \Pi_{applicantid,\ shelterid}(applications \bowtie animals)$$

$$spammers \leftarrow appinfo \div shelterinfo$$

$$\Pi_{applicants.name,\ applicants.\ phone}(\sigma_{spammers.applicantid = applicants.applicantid}(spammers \times applicants))$$

7. Rocket Science? Through some recent ground-breaking marketing research, surveys show that animal shelters offering pet rocks result in decreased trust from potential applicants in the legitimacy of the shelter. Write the series of relational algebra operations necessary to remove all records that refer to pet rocks (where *animal_type* is "Rock") from all tables in the database. Don't worry, they'll go to good homes.

$$applications \leftarrow applications - \Pi_{appid,\ applicantid,\ animalid,\ status,\ applicationdate}(\sigma_{animaltype = "Rock"}(applications \bowtie animals))$$

$$animals \leftarrow animals - \sigma_{animaltype = "Rock"}(animals)$$