Midterm1 Solutions

Question 1 (10 points)

(1) For each physician, list their id and name along with the total of their prescriptions.

```
SELECT P1.physician_id, physician_name,
      sum(P2.unit_price * quantity) as total_prescriptions
FROM exphysician P1 Join exprescription P2 on
      P1.physician_id = P2.physician_id
GROUP BY physician_id, physician_name;

or

SELECT P1.physician_id, physician_name,
      sum(P2.unit_price) as total_prescriptions
FROM exphysician P1 Join exprescription P2 on
      P1.physician_id = P2.physician_id
GROUP BY physician_id, physician_name;
```

(2) For each manufacturer, list the name and the average price of their computers.

```
SELECT man_name,avg(price) as avg_price
FROM excomputer C, exmanfacturer M
WHERE M.man_id = C.man_id
GROUP BY man_name;
```

(3) For each drug, list its name, the number of distinct patients the drug has been prescribed to, and total amount all patients have paid for the drug.

```
SELECT drug_name, COUNT(DISTINCT patient_id) n_patient,
      sum(unit_price*quantity) total_price
FROM exprescription pr, exdrug dr
WHERE pr.drug_id = dr.drug_id
GROUP BY pr.drug_id;

or

SELECT drug_name, COUNT(DISTINCT patient_id) n_patient,
      sum(unit_price) total_price
FROM exprescription pr, exdrug dr
WHERE pr.drug_id = dr.drug_id
GROUP BY pr.drug_id;
```

(4) For each manufacturer, list the manufacturer name along with the number of cpu and ram-size options they offer in their computers. Each cpu type should be counted only once even if the manufacturer use it in multiple computers (same thing for ram size).

```
SELECT man_name, COUNT( DISTINCT ram ) AS ram_options,
      COUNT( DISTINCT ( Cpu ) ) AS cpu_options
FROM excomputer EC, exmanfacturer EM
WHERE EC.man_id = EM.man_id
GROUP BY man_name;
```

Question 2 (11 points)

(1) List customer id and computer price for customers who purchased the cheapest computer. Eliminate duplicates in your result.

```
SELECT distinct cus_id, price
FROM excomputer C join excustomer_computer CC on C.comp_id = CC.comp_id
WHERE price = (SELECT MIN(price) From excomputer);
```

(2) List manufacturer name and ram size for manufacturers who offer the largest ram size in their computers. Eliminate duplicates in your result.

```
SELECT distinct man_name, ram
FROM excomputer C, exmanfacturer M
WHERE M.man_id = C.man_id and ram = (Select max(ram) from excomputer C);
```

(3) List physician id, name, and drug price for physicians who prescribed the most expensive drug (use unit_price attribute when comparing drug prices). Eliminate duplicates in your result.

```
SELECT Distinct PH.physician_id, physician_name, unit_price
FROM exphysician PH join exprescription PR
     on PH.physician_id = PR.physician_id
WHERE unit_price = (select max(unit_price) from exprescription);
```

(4) List drug id for drugs prescribed by the oldest physician(s) (i.e., physicians who have the largest age). Include drug id, physician name, and physician age in your results. Eliminate duplicates in your result.

```
SELECT DISTINCT drug_id, PH.physician_id, physician_name, age
FROM exphysician PH
     JOIN exprescription PR ON PH.physician_id = PR.physician_id
WHERE age = ( SELECT MAX( age ) FROM exphysician );
```

Question 3 (11 points)

(1) For each cpu type, show the average price from each manufacturer (list cpu, man_id, and average price). Include only manufacturers whose average for a cpu type is less than or equal the overall average price of that cpu type (average considering all computers from all manufacturers).

```
SELECT co.cpu, co.man_id, avg(price) average
FROM excomputer co
GROUP by co.cpu, co.man_id
HAVING avg(price) <= (select avg(price) from excomputer where co.cpu = cpu);
```

(2) List drug id, drug name, and the number of distinct patients the drug has been prescribed to. Include only drugs prescribed to everyone in the prescriptions table (i.e., all patients who already have prescriptions).

```
SELECT d.drug_id, drug_name,
      count(distinct patient_id) as num_of_distinct_patients
FROM exdrug d, exprescription p
WHERE d.drug_id = p.drug_id
GROUP BY d.drug_id
HAVING count(distinct patient_id) = (select count(distinct patient_id) from
exprescription);
```

(3) For each physician, show the number and highest age of their patients (list physician_id, physician_name, number of distinct patients, highest age of the physician's patients). Include only physicians who worked with patients who have the highest age in the data set.

```
SELECT phy.physician_id, phy.physician_name,
      count(distinct pre.patient_id) n_patients,
      max(pat.age) max_patients_age
FROM exprescription pre, exphysician phy, expatient pat
WHERE pre.physician_id = phy.physician_id and pre.patient_id = pat.patient_id
GROUP BY phy.physician_id, phy.physician_name
HAVING max_patients_age = (select max(age) from expatient);
```

(4) List the manufacturers which offer the largest ram size in the data set. Include the man_id, man_name and the largest ram size the manufacturer offers in its computers.

```
SELECT man.man_id, man_name, max(distinct com.ram) largest_ram
FROM exmanfacturer man, excomputer com
WHERE man.man_id = com.man_id
GROUP BY man.man_id, man_name
HAVING max(distinct com.ram) = (select max(distinct ram) from excomputer);
```

(5) List customers who purchased at least half of the computers in the store. Include the customer id, name, and the number of computers the customer has purchased.

```
SELECT c.cus_id, cus_name, count(comp_id) as no_purchased_computers
FROM excustomer c join excustomer_computer cc
    on c.cus_id = cc.cus_id
GROUP BY c.cus_id, c.cus_name
HAVING count(distinct comp_id) * 2 >= (select count(*) from excomputer)
```

Question 4 (5 points)

(1) Using relational algebra, list zipcodes for customers who purchased a computer with RAM size larger than 64 GB. Write your relational algebra on paper (or use your favorite tool), take a photo (or scan) your solution, and upload the solution using the add-file button below.

$$\pi_{zipcode}(\sigma_{RAM>64}(excustomer \bowtie excustomer\_computer \bowtie excomputer))$$

(2) Using relational algebra, list manufacturer names who have computers with prices greater than 1200. Hint: you can use natural join for a shorter query. Write your relational algebra on paper, take a photo (or scan) your solution, and upload the solution using the button below.

$$\pi_{man\_name}(exmanfacturer \bowtie (\sigma_{Price>1200}(excomputer)))$$

(3) Using relational algebra, list manufacturer names who have computers with prices greater than 1200. Write your relational algebra on paper (or use your favorite tool), take a photo (or scan) your solution, and upload the solution using the add-file button below.

$$\pi_{man\_name}(exmanfacturer \bowtie (\sigma_{Price>1200}(excomputer)))$$

(4) Using relational algebra, list the names of the customers who have purchased a computer with an I7 cpu. Write your relational algebra on paper (or use your favorite tool), take a photo (or scan) your solution, and upload the solution using the add-file button below.

$\pi_{cus\_name}(excustomer \bowtie excustomer\_computer \bowtie \sigma_{cpu='i7'}(excustomer))$

(5) Using relational algebra, list the names of physicians who have prescribed drugs cheaper than 20 dollars. Write your relational algebra on paper (or use your favorite tool), take a photo (or scan) your solution, and upload the solution using the add-file button below.

$\pi_{physician\_name}(\sigma_{unit\_price<20}(exprescription \bowtie exphysician))$

(6) Using relational algebra, list the names of drugs which have been prescribed to patients older than 35 years old. Write your relational algebra on paper (or use your favorite tool), take a photo (or scan) your solution, and upload the solution using the add-file button below.

$\pi_{drug\_name}(exdrug \bowtie exprescription \bowtie (\sigma_{age>35}(expatient)))$

(7) Using relational algebra, list the names of patients who have been prescribed Ampicillin. Write your relational algebra on paper (or use your favorite tool), take a photo (or scan) your solution, and upload the solution using the add-file button below.

$\pi_{patient\_name}(\sigma_{drug\_name='Ampicillin'}(ex\ drug) \bowtie exprescription \bowtie expatient)$

Question 10 (10 points)

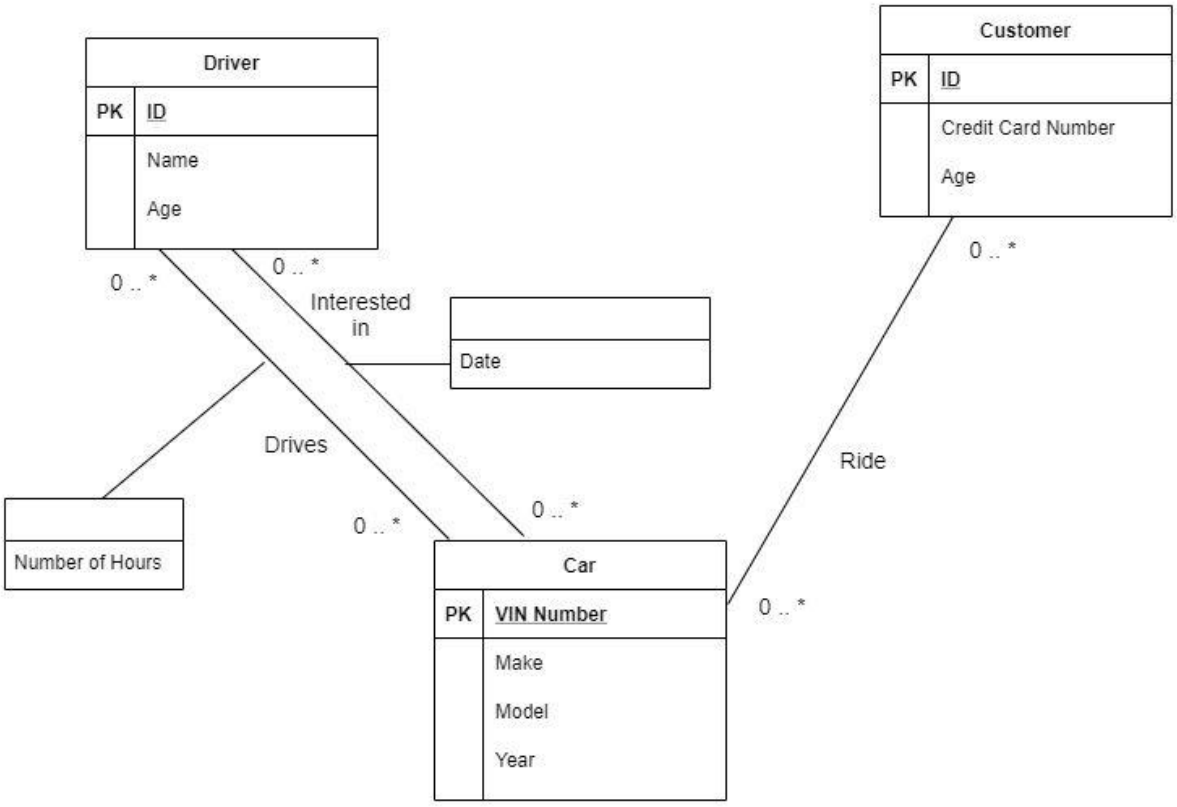
(1) SUBER is a TAXI company whose motto is "why ride UBER if you can ride SUBER." Knowing that you are a boilermaker taking CS348, SUBER wants you to draw an ERD for its new driver-customer information system. The ER Diagram covers following information.

- A driver is identified by driver id and has a name and age. A customer is also identified by customer id and has age and credit card number. A car however is identified by VIN number and has a make, model, and year.

- SUBER records the number of hours each driver has driven a specific car in its fleet. For example, John Smith has driven a Toyota Camry 200 hours then switched to driving a Ford Taurus for 250 hours. Sally Tucker was given the Toyota Camry that John has left. Sally drove the Camry for 500 hours. SUBER drivers may also submit their interest to drive a car or more in the company fleet. The company keeps record of the date the driver declared interest in driving a specific car. This date can be used to decide which driver can get a car if multiple drivers have interest in it.
- SUBER also keeps track of which cars a specific customer has ridden. For example, David Shapiro has ridden both the Camry and the Taurus mentioned above while Kristin Macray has ridden only the Taurus.

For the scenario above, use only three entities: Driver, Car, and Customer. Use only the description and the example data provided to decide what attributes, relationships, and cardinalities you include in your ERD. If the minimum cardinality is not clarified in the text, then you can use either 0 or 1. Draw your ERD on paper (or using your favorite tool) and upload a photo of your ERD using the add-file button below. Use the textbox, to write any notes you have.

Answer Key



(2) Knowing that you are a boilermaker taking CS348, the Meijer grocery chain asked you to draw an ERD for its new produce-department information system. In return, Meijer offered unlimited fruits and vegetables for you until you graduate. The ER Diagram covers following information.

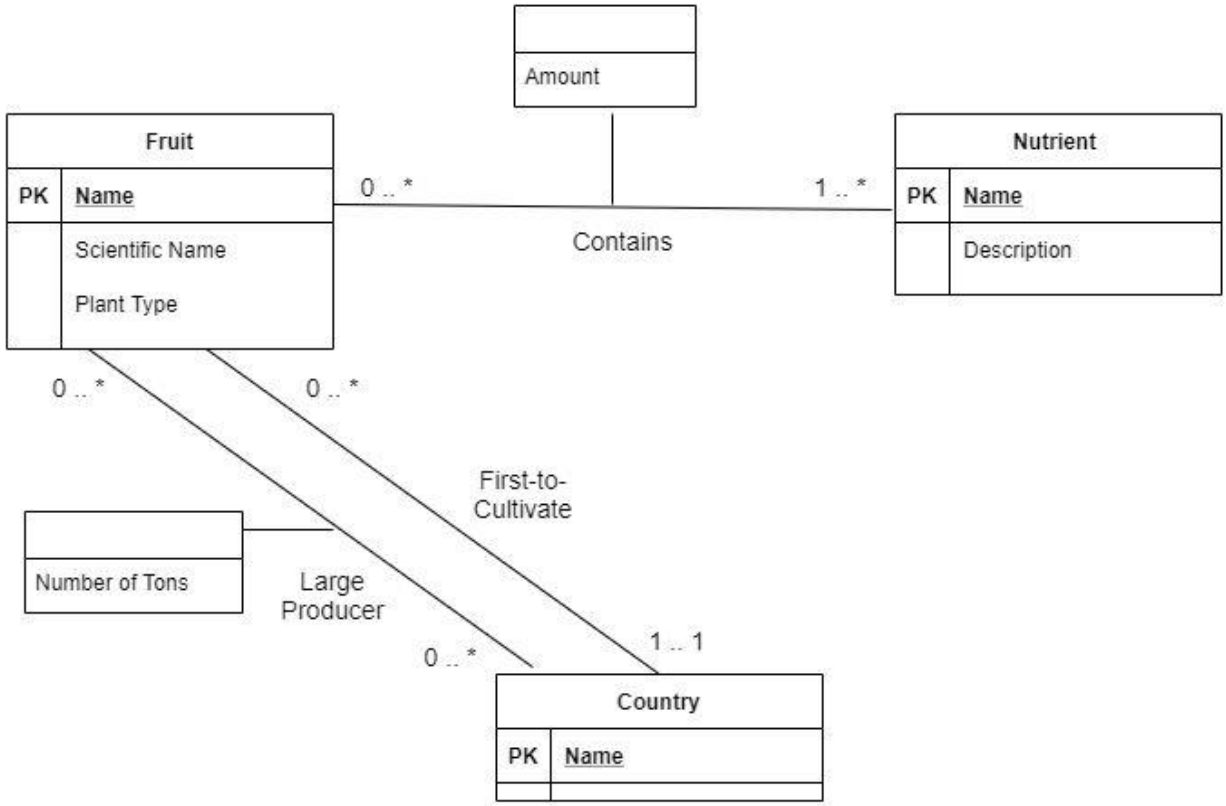
For the scenario below, use only three entities: country, fruit, and nutrient.

- A fruit information includes name, scientific name, and plant type (e.g., tree or bush). A fruit name is unique.
- A nutrient (e.g., Vitamin C and Vitamin D) has a name and description that explains the benefits of the nutrient. Each fruit contains a different number of nutrients with different amounts. For example, mango contains 100 mg of vitamin C and 3 mg of

vitamin K while Kiwi contains 92 mg of vitamin C and 2 mg of vitamin E. A fruit has an at least one nutrient.

- A country is identified by name. A country can be among the large producers of a certain fruit. In this case, the information includes the number of tons produced per year. A country can also be the first to cultivate a fruit. For example, orange was first cultivated in China; currently, Brazil, India, and China are among the largest producers with 18, 4, and 3 million tons per year, respectively. China is also the largest producer for Kiwi with 2 million tons per year.

For the scenario above, use only three entities: country, fruit, and nutrient. Use only the description and the example data provided to decide what attributes, relationships, and cardinalities you include in your ERD. If the minimum cardinality is not clarified in the text, then you can use either 0 or 1. Draw your ERD on paper (or using your favorite tool) and upload a photo of your ERD using the add-file button below. Use the textbox, to write any notes you have.



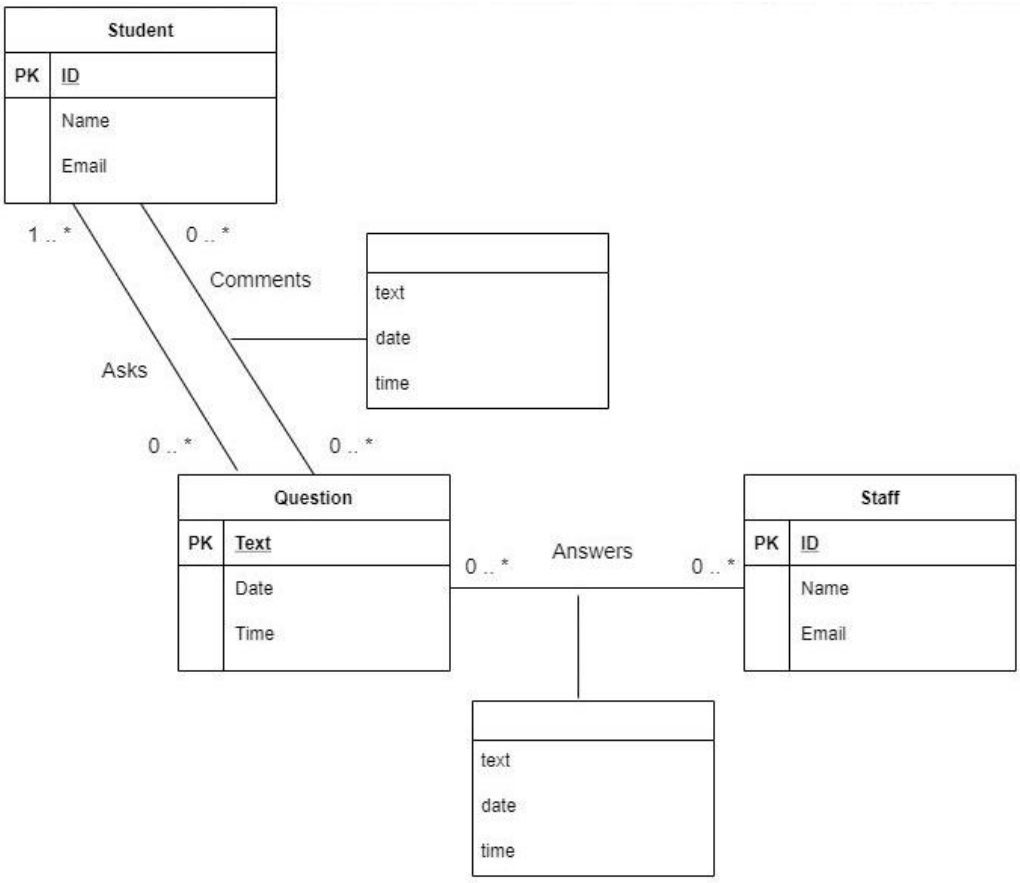
(3) Campuswire has abandoned its old database design that contributes to the company frequent down time. Knowing that you are a boilermaker taking CS348, the company CEO

felt so sorry for the inconvenience their site has caused. The CEO is asking for your help to draw an ERD for its new question-answering database. The ER Diagram covers following information.

For the scenario below, use only three entities: Student, Staff, and Question.

- A staff member and a student are both identified by id and they have names and email addresses. A student can ask questions where the system saves the text of each question and the date and time the question was asked. A new feature that Campuswire wants to add is allowing multiple students to ask the same question (e.g., group members asking a question about a bug in their code). A student can also have multiple comments on questions in Campuswire. A question may get comments from different students. The system records the comment text and the date and time the comment is posted.
- A staff member can answer multiple questions in Campuswire (for simplicity, a staff member can answer a specific question only once). Bringing different perspectives, multiple staff members can answer the same question to enrich the discussion. The system keeps track of the answer text and the date and time the answer was posted.

For the scenario above, use only three entities: Student, Staff, and Question. Use only the description and the example data provided to decide what attributes, relationships, and cardinalities you include in your ERD. If the minimum cardinality is not clarified in the text, then you can use either 0 or 1. Draw your ERD on paper (or using your favorite tool) and upload a photo of your ERD using the add-file button below. Use the textbox, to write any notes you have.



Question key was not clarified in the text. Therefore, it can be one of the following:
-text
-text, date, time ((text, date) or (text, time) are also acceptable)
- Adding an ID for question (this is likely the way campusewire identify questions).

Question can be represented as a weak entity where student is the strong entity (even though in the description a question can be asked by multiple students, which contradicts the requirement that a weak entity has to be related to only one strong entity).