CS127 Homework #2

Due: September 25th, 2019 11:59 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.

Some notes before beginning:

1. Tuple relational calculus is difficult to read. It would be extremely helpful for all of us (you the writer, and we the graders) if you followed the following style when writing it (from the book):

$$
\begin{aligned}
& \{t \mid \exists r \in student(r[ID] = t[ID]) \land \\
&\hspace{1cm}(\forall u \in course(u[dept\_name] = \text{“}CS'' \Rightarrow \\
&\hspace{2cm} \exists s \in takes(t[ID] = s[ID] \\
&\hspace{3cm} \land s[course\_id] = u[course\_id]))\}
\end{aligned}
$$

   Note that when the parentheticals become unwieldy, move to the next line and indent. You don't have to write it in LaTeX. However, if you choose to do so, this can be accomplished using the following code (note the `hspace` command to add some space):

```
\begin{align*}
    & \{t \mid \exists r \in student(r[ID] = t[ID]) \land \\
    &\hspace*{1cm}(\forall u \in course(u[dept\_name] = ‘‘CS’’ \Rightarrow \\
    &\hspace*{2cm} \exists s \in takes(t[ID] = s[ID] \\
    &\hspace*{3cm} \land s[course\_id] = u[course\_id]))\}
\end{align*}
```

2. When you draw your ER diagrams and draw your relational schema, use the conventions found in the book (also found in the appendices). We also recommend using some software to draw them so that you can make edits. We recommend `draw.io` because it's, free and easy to use, and connects to your Google drive.

Warmup #1

Given the following relational database:

Nursery(<u>n_id</u>, n_name, address, state)
Tree(<u>t_id</u>, species_name, height, tree_type, leaf_type, years_to_maturity)
Sells(<u>n_id</u>, t_id, price)

1. Write a tuple relational calculus formula that returns the tree species name(s) with the longest time to maturity.
2. Find the names of nurseries who supply one or more trees that are 10 feet or taller.
3. Find the IDs of nurseries based in California who supply trees that are 10 feet or taller and have a price of more than 20 dollars.
4. Find all pairs of nursery IDs such that the nursery with the first ID charges more for some tree than the nursery with the second ID.
5. State what the following queries compute:
   (a) $\{t \mid \exists t_1 \in Tree(t[t\_id] = t_1[t\_id] \wedge t[species\_name] = t_1[species\_name] \wedge \exists s \in Sells(t_1[t\_id] = s[t\_id] \wedge \exists n \in Nursery(s[n\_id] = n[n\_id] \wedge n[n\_name] = \text{“Johnny Appleseed”})))\}$
   (b) $\{n \mid \exists n_1 \in Nurseries(n[n\_id] = n_1[n\_id] \wedge n[n\_name] = n_1[n\_name] \wedge \exists s_1 \in Sells(s_1[n\_id] = n_1[n\_id] \wedge \forall s_2 \in Sells(s_1[price] \geq s2[price]))))\}$

Warmup #2

1. Define what it means for a query to be “safe”. (See textbook p. 244)
2. Determine whether the following queries are safe, and explain your answer. Be careful!
   - $\{t \mid \neg(t \in Trees) \wedge t \in Shrubs\}$
   - $\{t \mid \neg(t \in Trees) \vee t \in Shrubs\}$

Warmup #3

The purpose of this problem is to give you practice moving through the process of identifying entities and relationships, then creating E-R diagrams from entity/relationship sets, and then creating a database schema from the E-R diagram. Consider the following:

- A person has an id, name, age, address, and phone numbers.
- A gardener has years of experience and species of trees that they cultivate.
- A tree is defined by its species type, and has a maximum height and years to maturity.
- Every gardener is a person, but not all people are gardeners.
- Every gardener specializes in cultivating at least one type of tree species, but there are some tree species are not cultivated by any gardeners.
- A person can have 0, 1, or 2 phone numbers.
- A person's address is divided into subparts: street_address, state, zip_code, country. Likewise, a person's name is divided into first_name, middle_name (which may be null), and last_name.

1. List the entity sets as they are given above with their respective attributes following them in parentheses. If you have redundant attributes, remove the redundancies. Underline the primary keys. For extra clarity, put multi-valued attributes in curly brackets, and group the parts of composite attributes with parentheses. (You may want to consult the textbook pp. 272-274).
2. What relationship sets represent the relationships between the entities?
3. Now, create an ER diagram representing the entity and relationship sets. Be sure to represent cardinalities, total vs. partial participation, primary keys and composite attributes. If you represent Gardener as a weak entity set, use the notation for weak entity sets.
4. Now translate your E-R Diagram into a database schema and write down your schema. Note primary keys and foreign keys. (Your schema will probably have multiple phone numbers per person, and even though that is not first normal form, that is OK for now, since we haven't yet covered database normalization in the lectures.)

Graded Problems

The following is a typical task you might see in the real world. Many people who are handling data haven't had training with designing databases or working with complex data. Sometimes, they end up storing their data in a single massive flat file. The graded problem sets uses the following scenario:

A small airline has a website from all over the world where users can buy their tickets. has stored all of its flight purchases in a CSV file. Your task is to transfer these data into a database to make life easier. They also have a "data dictionary", a list of the column headers and a description of what each column contains. This data dictionary is here: `https://bit.ly/2lUxW5D` (you'll need to sign-into your Brown Google account). Don't assume anything else besides hat the data dictionary contains because the airline has decided to keep their own internal conventions for what these data mean.

The database schema you design should be as general as possible. In addition to being used to record any of the worldwide purchases this small airline services, it will be used for a variety of tasks commonly used by the airline like:

- Determine that a passenger is not on two flights
- Find a flights for a passenger wanting to go from airport A to airport B
- Look up a group of passengers travelling together

The process of designing a schema is iterative. We'll walk through this process in the next two problems.

Problem 4

1. (15 points) Come up with a list of entities in the data and list them down here. Feel free to give entities an ID if one is not available and is necessary. Write your entities as follows:

   entity(<u>primary_key</u>, <u>primary_key</u>, attribute1, attribute2)

   Justify why these entities make sense with a sentence or two. As you go to the next parts, you might end up changing this list when you determine the relationships between these entities.

2. (15 points) Draw an ER diagram using these entities. To do this, you'll need to determine the relationships between the entities (the entity set) you wrote up in the previous part. Note that you should be consistent with the entities you wrote above. This process might help you rethink your list of entities. *See Appendix A for the notation you should use in this homework.*

   Justify each of the relationships between the entities. For example, we expect that you will explain why the relationship is a one-to-many or many-to-many relationship, whether it relies on some other entity for a primary key (i.e. one entity is a weak entity), and if there is an attribute, why this relationship attribute exists.

   Here are some hints you might want to consider:

   (a) Consider that a passenger and a customer may be the same *person*

   (b) Consider each of the entities in your list first. Then, instead of drawing all relationships at once, define the relationships between closely related entities. For example, there seems to be a subset of entities having to do with people (e.g. customers and passengers), another having to do with purchases (e.g. purchase amount, another having to do with aircraft, and another having to do with routes. Determine the relationships within these subsets first.

   (c) Now consider how these subsets of entities relate to other subsets (e.g. the subset having to do with customers/passengers and the subset having to do with purchases). What entities or relationships tie these subsets together?

(d) The route column is tricky. Consider a route from BOS → PHL → SFO → LAX This route has two stops (PHL and SFO). This is also composed of the following direct routes: BOS → PHL, and PHL → SFO, and SFO → LAX. It also has the following *routes with one stop*: BOS → PHL → SFO, and PHL → SFO → LAX.

3. (15 points) From your ER diagram, draw a relational schema. The goal here is to be consistent with your ER diagram. Justify how you decided to represent each of the relationships in your ER diagram in your schema. For example, did you make the relationship, part of a primary key, a foreign key, or a separate relation and why? Identify the primary keys and foreign keys in your schema and make sure you define the types for each of your attributes. *See Appendix B for an example schema diagram and the notation you should use in this homework.*

Problem 5

Now that you've made your ER diagram and relational schema, it's time to see if users can use your database to answer some questions. If you realize that it isn't quite able to answer the question, or if you think that there's a better way to lay-out the schema, feel free to change your answers in the previous problem.
Here are two common queries that the airline might want to ask your database. Write each of them down in relational algebra using the schema you made. (10 points each)

1. Find flights from Providence (PVD) to Los Angeles (LAX) on September 30, 2019, with at most 1 stop, with an available seat in the "business" section of the airplane.
2. A group of 3 passengers missed their connection. Given one of the passengers' records, find a route to their destination where the entire group can fly (regardless of the section of the airplane they sat in).

And, for the sake of practice, write these down using tuple relational calculus (15 points each).

Problem 6

The teaching staff takes your input very seriously and very much to heart and we hope to gain insight on your experience so far.
These questions are our attempt to gather feedback from the entire class while the semester is just beginning. Because the grading process is single-blind (we don't know who we are grading), please do not hesitate to provide whatever <u>constructive</u> criticism you might have. These questions are worth 5 *free* points total.

1. List 3 points the class is doing well in or needs improvement. Provide a one sentence description for each. (1 point for each of the points, 3 points total)
2. What you would like to learn in this class (e.g. data structures, concurrency, other data models, "Big Data" processing)? (1 point)
3. What would you like to see from the teaching staff (beyond what we are already doing that needs improvement)? (1 point)

Appendix A: ER Diagram symbols

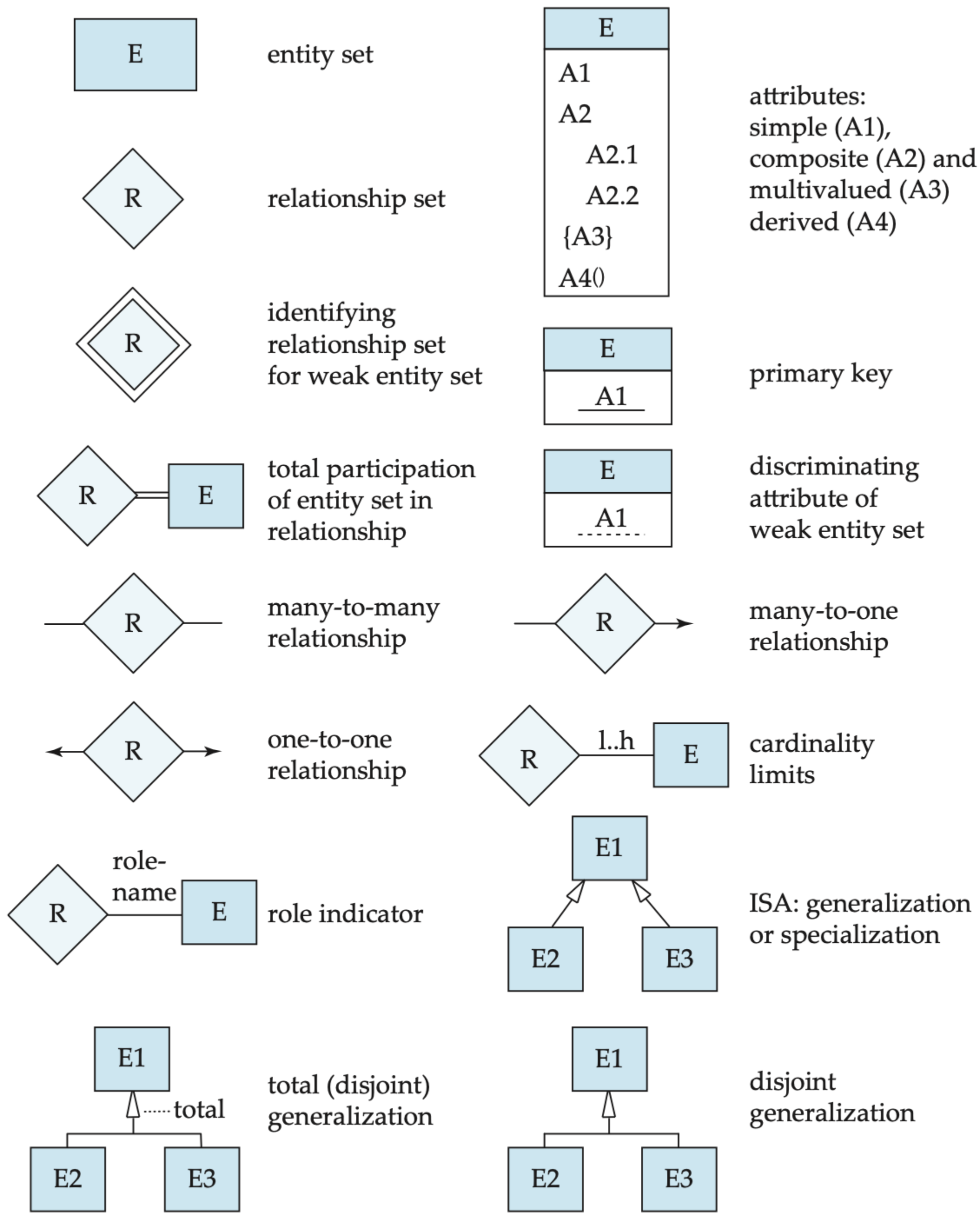

Figure 7.24 Symbols used in the E-R notation.

Appendix B: Schema Diagram symbols

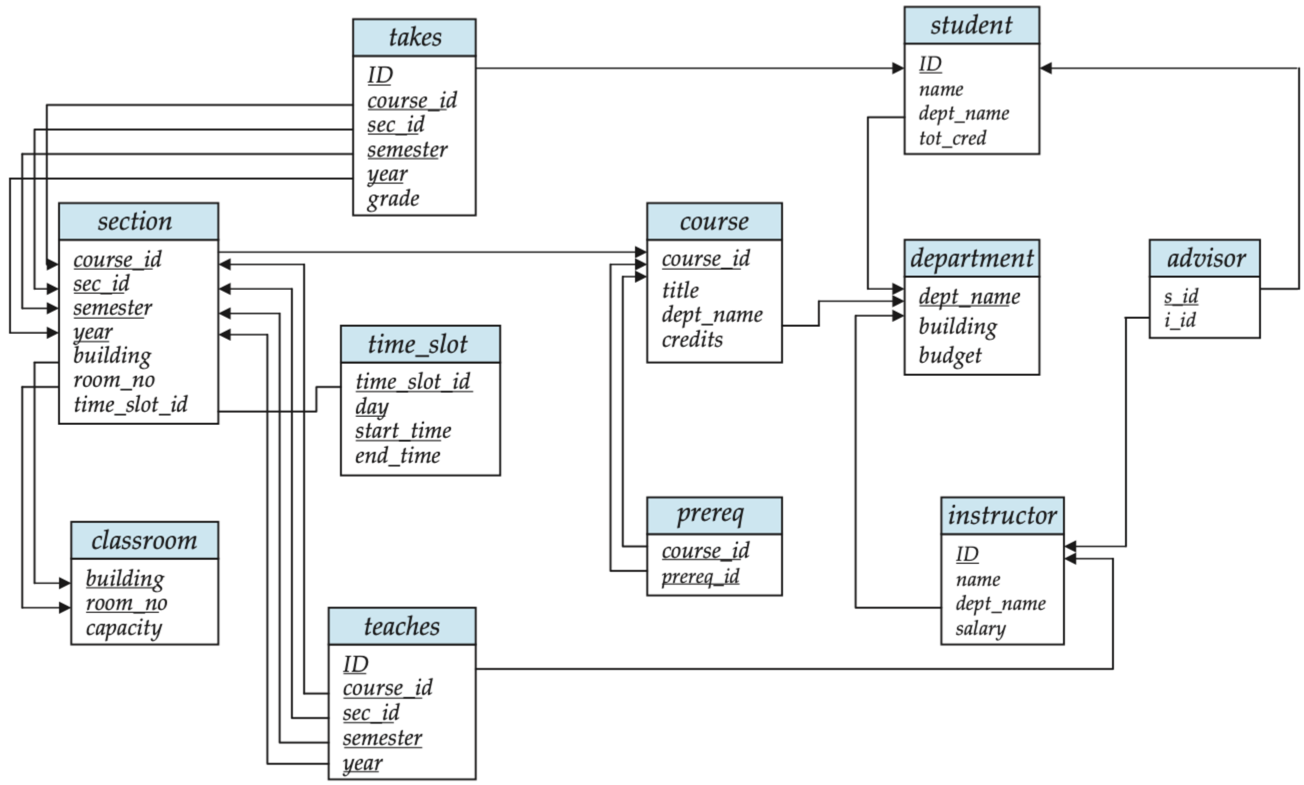

Figure 2.8 Schema diagram for the university database.

Note the following:

1. One box in the diagram is a relation. The box is topped by the name of the relation. The contents of the box are the relation's attributes. You don't have to make it look exactly like this but it the relation's name and attributes needs to be clear.
2. Underline primary keys.
3. Arrows from one attribute to another indicate a foreign key. If attribute A in a relation references another attribute B in the same relation, you can draw an arrow from A back to attribute B in the same relation.