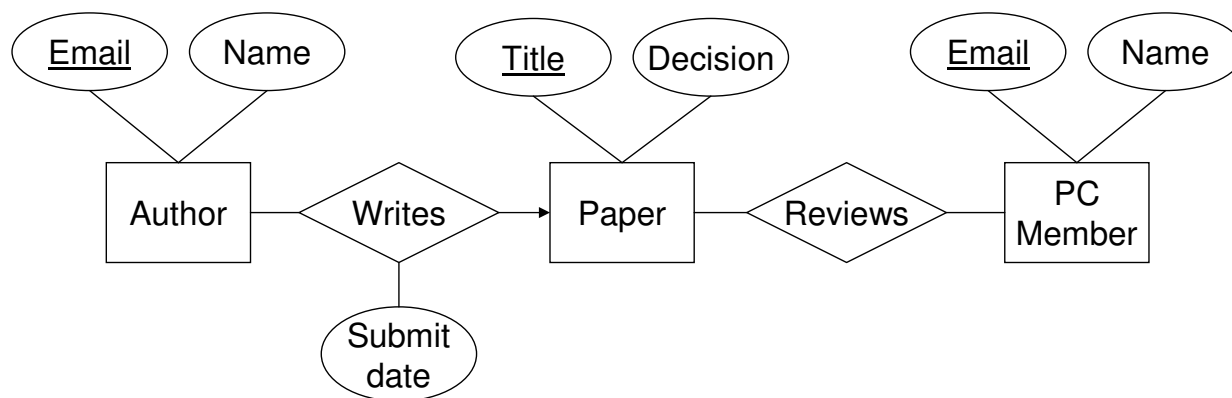


Time will be tight. If you spend more than the recommended time on any question, **go on to the next one**. If you can't answer it in the recommended time, you are either going in to too much detail or the question is material you don't know too well. You can skip one or two parts and still demonstrate what I believe to be an A-level understanding of the material.

Note: It is okay to abbreviate in your answers, as long as the abbreviations are unambiguous and reasonably obvious (e.g., use PE and AE for PC Member Email and Author Email in the first question.)

1 Data Modeling (15 minutes, 14 points)

Given the following ER diagram, for a database to manage papers submitted to a conference:



1.1 Convert to Relational (3 minutes, 3 points)

Give relations corresponding to the ER diagram. Don't worry about normalization - just give the conversion from the diagram.

1.2 Give Functional Dependencies (3 minutes, 6 points)

What functional dependencies hold on this data? For each dependency, note briefly (one sentence) how you derived it (is it inherent in the ER diagram? Something you think should be true?)

1.3 Modifications (3 minutes, 2 points)

Currently “people” are referenced in two different ways: As PC Members, and as Authors. It is quite possible that someone could be both a PC member and an Author, giving us redundant information. Modify the above diagram to eliminate this redundancy.

1.4 Constraints (4 minutes, 3 points)

Come up with at least one other constraint you feel should hold on the data. Can you represent this in the ER model? If so, sketch how, if not, suggest why not.

2 Relational Design (20 minutes, 14 points)

You are to develop a relational schema for a database to be used for conference registration and planning. The goal is to represent when and where talks are being held, who is speaking and attending each, and contact information of the people involved.

You are given the following data:

UniversalRelation(Speaker, SAddress, SEmail, Participant, Address, Email, Room, Hour, Talk)

the following functional dependencies (use capitalized letters above for abbreviations):

$$\begin{array}{ll} E \rightarrow P & SE \rightarrow S \\ P \rightarrow E & S \rightarrow SE \\ P \rightarrow A & S \rightarrow SA \\ T \rightarrow RH & T \rightarrow S \\ RH \rightarrow T & \end{array}$$

and the additional constraint that every Speaker must be registered as a Participant, with the Speaker email (SE) and address (SA) the same as the address used when they registered as a participant (E,A).

2.1 Boyce-Codd Normal Form (7 minutes, 5 points)

Give a BCNF decomposition that maintains lossless join.

2.2 4NF

Some of the talks may be day-long or multi-day tutorials. Thus there may be several rooms and hours for a single talk, however each has the same speaker and participants. The functional dependencies above prevent this.

2.2.1 Dependencies (3 minutes, 3 points)

Show what functional dependencies need to be removed, and what functional or multivalued dependencies need to be added, to support such multi-room/hour talks.

2.2.2 Schema (4 minutes, 3 points)

Give a good schema for your new collection of dependencies.

2.3 Lossless join (4 minutes, 3 points)

Give a query in relational algebra that reconstructs the original relation *UniversalRelation* from your decomposition of either question 2.1 or 2.2.2 (note which you use).

3 Queries (15 minutes, 10 points)

You will need to answer queries on the following schema, used to capture event information, registrations, and payments at the conference (for this conference, each event is registered for and charged separately.)

```
create table Receipts (
  Participant varchar(40),
  Amount      numeric
);

create table Registrations (
  Participant varchar(40),
  Event       char(5) references EventInfo(Event),
  primary key (Participant, Event)
);

create table EventInfo (
  Event   char(5) primary key,
  Speaker varchar(40),
  Room    char(5),
  Time    timestamp,
  Price   numeric
);
```

Answer each of the following queries. If you do one in SQL, and one in relational algebra, you get an extra point.

3.1 Simple query (5 minutes, 3 points)

Produce a schedule for participant *Chris Clifton* showing what event, where, and when. For full credit, the events should be listed in order of the time that they occur.

3.2 Reports (5 minutes, 3 points)

Produce a report showing the total earned by each event (i.e., number of participants * price).

3.3 Understanding a Query (3 minutes, 3 points)

Describe briefly what the following queries means (i.e., what the values in the result correspond to in real-world terms).

```
select O.Participant, O.Amount-R.Amount Total
from (select R.Participant, sum(E.Price) Amount
      from Registrations R, Eventinfo E
      where R.Event = E.Event
      group by R.Participant) O,
      (select Participant, sum(Amount) Amount
      from Receipts
      group by Participant ) R
where R.Participant = O.Participant;
```