CSCE 520, Midterm Exam II, Thursday November 8, 2018

Do all problems. You have 75 minutes. Open book, open notes, no electronic devices of any kind. There are 110 points total in the exam, of which 100 points count as full credit with the rest extra credit. Undergraduates get a free 10-point boost.

You do not need to show your work to get full credit for a correct answer, but if your answer is incorrect, showing your work may earn you more partial credit.

Some problems below refer to the following relational database schema, which is similar to the one in the last midterm:

```
Airport(name, code, city, capacity)
Serves(apCode, alName)
Airline(name, headqtrs)
Flight(alName, fltNo, depCode, arrCode, depDate, numPass)
```

Here are the SQL commands used to create the tables, above, *mutatis mutandis* from before:

```
CREATE TABLE Airport (
    name         VARCHAR(20),
    code         CHAR(3)       PRIMARY KEY,
    city         VARCHAR(20),
    capacity     INT
);
CREATE TABLE Serves (
    apCode  CHAR(3)      REFERENCES Airport(code),
    alName  VARCHAR(20)  REFERENCES Airline(name)
);
CREATE TABLE Airline (
    name      VARCHAR(20)   PRIMARY KEY,
    headqtrs  VARCHAR(20)
);
CREATE TABLE Flight (
    alName         VARCHAR(20)   REFERENCES Airline(name),
    fltNo          INT,
    depart         CHAR(3)       REFERENCES Airport(code),
    arrive         CHAR(3)       REFERENCES Airport(code),
    depDate        DATE,
    numPass        INT,  /* Number of passengers on board */
    PRIMARY KEY (alName, fltNo, depDate)
);
```

We will also have occasion to refer to this specific instance of the database (as before):

Airport: (sorted by *code*)

name	*code*	*city*	*capacity*
Hartsfield	ATL	Atlanta, GA	1260
Logan	BOS	Boston, MA	200
Columbia	CAE	Columbia, SC	30
De Gaulle	CDG	Paris, FR	800
Douglas	CLT	Charlotte, NC	420
Metro	DTW	Detroit, MI	250
Newark	EWR	Newark, NJ	600
Frankfurt	FRA	Frankfurt, GR	2000
Kennedy	JFK	New York, NY	1659
Los Angeles	LAX	Los Angeles, CA	1350
LaGuardia	LGA	New York, NY	1700
Gatwick	LGW	London, UK	1000
Heathrow	LHR	London, UK	700
Midway	MDW	Chicago, IL	50
Narita	NRT	Tokyo, JP	800
O'Hare	ORD	Chicago, IL	1490
Orly	ORY	Paris, FR	650
Sheremetyevo	SVO	Moscow, RU	320

Serves: (sorted first by *code*)

apCode	*alName*
ATL	Delta
BOS	American
CAE	United
CDG	Air France
DTW	Northwest
EWR	Continental
FRA	Lufthansa
JFK	Delta
JFK	US Airways
LAX	Virgin Pacific
LGA	Delta
LGA	US Airways
LGW	British Airways
LGW	US Airways
LHR	British Airways
MDW	Midway
NRT	Northwest
ORD	American
ORD	United
ORY	Air France
ORY	Delta

Airline: (sorted by *name*)

name	*headqtrs*
Aeroflot	Moscow, RU
Air France	Paris, FR
American	Dallas, TX
British Airways	London, UK
Continental	Jersey City, NJ
Delta	Atlanta, GA
Lufthansa	Frankfurt, GR
Midway	Chicago, IL
Northwest	Detroit, MI
United	Chicago, IL
US Airways	Pittsburgh, PA
Virgin Pacific	Seattle, WA

Flight: (sorted by *alName* then *fltNo* then *depDate*)

alName	*fltNo*	*depart*	*arrive*	*depDate*	*numPass*
Aeroflot	80	SVO	JFK	2017-06-15	282
American	704	CLT	FRA	2018-09-23	218
American	704	CLT	FRA	2018-09-24	191
American	705	FRA	CLT	2018-09-27	230
American	705	FRA	CLT	2018-09-28	210
Delta	985	ATL	LGA	2018-04-01	56
Delta	986	LGA	ATL	2018-03-17	80
Lufthansa	501	FRA	JFK	2018-06-03	275
Lufthansa	870	LHR	FRA	2017-12-25	49
Northwest	106	DTW	NRT	2017-01-23	279
Northwest	106	DTW	NRT	2018-01-23	290
United	413	CAE	ORD	2018-07-14	79
United	415	ORD	CAE	2018-07-17	82

1. (5 points each; 20 points total) For each of the four functional dependencies (FDs (a)–(d)) given below, say explicitly whether or not the instance given on the previous page satisfies the given FD. If the FD is violated, give a pair of tuples from the table that witness the violation.

   a) `Airport.city -> Airport.name`
   b) `Flight.fltNo -> Flight.alName`
   c) `Airport.name -> Airport.code`
   d) `Flight.depart Flight.arrive -> Flight.alName`

2. (15 points) Consider a relational schema $R(A, B, C, D)$ with set S of functional dependencies

$$AB \to C$$
$$C \to D$$
$$D \to B$$

   Decompose R fully into BCNF. Your decomposition should be efficient. You only need to give the final tables, that is, you may omit the tree.

3. (5 points) Let an instance of $R(A, B, C)$ have the following tuples: $(1, 3, 4)$, $(1, 3, 5)$, $(2, 4, 5)$, $(2, 4, 6)$, and $(2, 5, 6)$. Give the fewest possible tuples to add to this instance so that it satisfies the multivalued dependency $A \twoheadrightarrow C$.

4. (10 points each; 30 points total) What is returned by each of the following SQL queries, applied to the instance on page 2? Give your answer in tabular form, including column headers.

   (a)
```
SELECT headqtrs
FROM Airline
WHERE name IN (
    SELECT alName FROM (
        SELECT alName, fltNo, MIN(numPass)
        FROM Flight
        GROUP BY alName, fltNo
        HAVING MIN(numPass) >= 200) N
    WHERE alName <> 'Lufthansa');
```

   (b)
```
SELECT fltNo, depDate
FROM Flight F
WHERE arrive NOT IN (
    SELECT apCode FROM Serves S
    WHERE S.alName = F.alName);
```

   (c)
```
SELECT capacity
FROM Airport
WHERE city IN (SELECT headqtrs FROM Airline) AND code IN (
    SELECT depart FROM Flight
    WHERE depDate >= '2018-06-01');
```

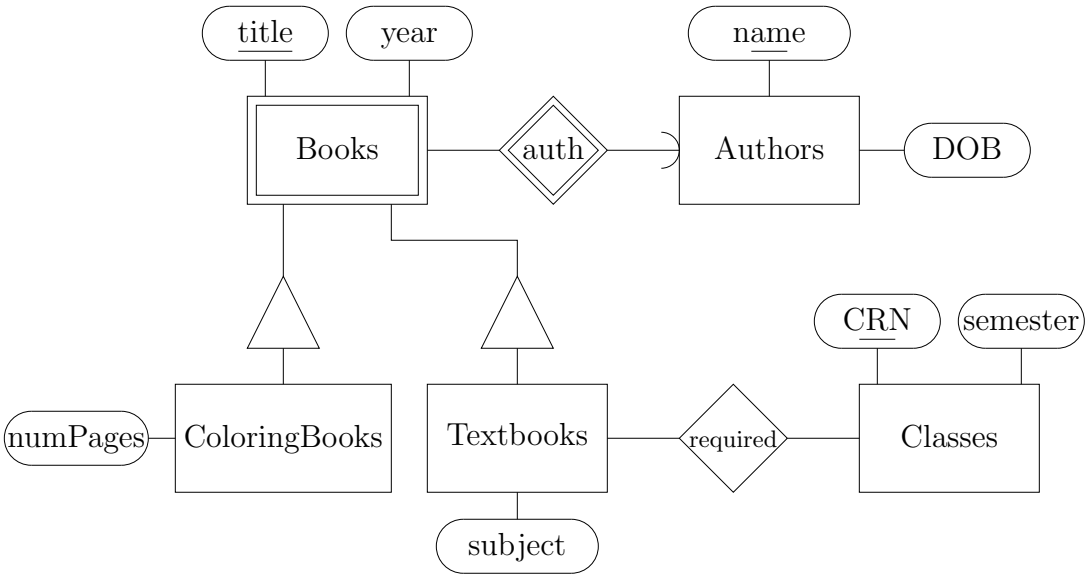
5. (15 points each; 30 points total) Render the following queries in SQL so that no `FROM` clause mentions more than one relation (this will force you to use subqueries, and the same restriction applies to these). Your query should be correct for any instance of the schema on page 1, not just the instance given on page 2. There are usually several correct answers for each; make yours reasonably simple and economical.

   For both of these queries, you may assume that all flights arrive the same day they depart.

   (a) (10 points) Those airports (given by their codes) that saw more departing passengers than arriving (totals for all flights) today (i.e., on November 8, 2018). Include the total number of departing passengers (but not arriving) with each airport code you show. [November is the 11th month of the year.]

   (b) The airlines (given by name) that had no flights (of that airline) arriving during the month of July, 2018 at any of the airports that serve them. [For example, in the instance on page 2, United would *not* be on the list, because there was a United flight arriving at ORD (which serves United) during the month of July 2018.] (Note: It is possible that an airline is not served by any airport, in which case it should definitely show up in your list.)

6. (15 points) Consider the following E/R diagram:



   Using the E/R viewpoint for "is-a" hierarchies, convert this diagram into a relational schema. Give your relational schema informally—a series of table names with attribute names in parentheses for each. Underline attributes that make up the primary key for each relation, and describe any other constraints separately in prose. Choose reasonable names for attributes, and make your decomposition as economical as possible.