CSCE 520, Midterm Exam I
Thursday October 5, 2017

Some problems refer to the following relational database schema, related to music (and not necessarily realistic):

```
Genre(code, name)
Song(title, genreCode, cDate)
Composer(name, DOB)
Writes(composerName, songTitle)
Performance(perfName, songTitle, length, perfDate, recorded)
```

Here are the SQL commands used to create the tables, above (note: 'T' stands for TRUE and 'F' stands for FALSE:

```
CREATE TABLE Genre (
    code CHAR(4) PRIMARY KEY,
    name CHAR(20)
);
CREATE TABLE Song (
    title     VARCHAR2(255) PRIMARY KEY,
    genreCode CHAR(4) REFERENCES Genre(code),
    cDate     DATE
);
CREATE TABLE Composer (
    name VARCHAR2(50) PRIMARY KEY,
    DOB  DATE
);
CREATE TABLE Writes (
    composerName VARCHAR2(50) REFERENCES Composer(name),
    songTitle    VARCHAR2(255) REFERENCES Song(title)
);
CREATE TABLE Performance (
    perfName        VARCHAR2(50),
    songTitle       VARCHAR2(255) REFERENCES Song(title),
    length          INTEGER CHECK(length > 0),
    perfDate        DATE,
    recorded        CHAR(1) CHECK(recorded in ('T','F','?')),
    PRIMARY KEY (songTitle, performanceDate)
);
```

Note:

- Each song has a *title* that uniquely identifies it. It also has a *genreCode* (a code of up to three letters describing the genre, e.g., jazz, clas, rock, altr, ambt, hiho, cnty, ska, rb, etc., whose corresponding name is found in the Genre table) and a date of composition (*cDate*).
- Each composer has a *name*, which uniquely identifies her or him, and a *DOB* (date of birth).

- A composer may write several songs, and a song may be (co)written by any number of composers.
- Each performance is of a performer (a single person or an ensemble) performing a song and has a *length* in seconds and a date of performance. It may or may not be *recorded*. The key constraint requires that the same song is not performed twice on the same date.

1. (8 points each; 48 points total) Give expressions in relational algebra for the following queries:
   (a) the titles of all songs written by Cole Porter
   (b) the composer names and song titles of songs performed (at least once) by their composers
   (c) the performance dates and names of the performers of the song, "Anything Goes," performed in 2015 and recorded[1]
   (d) the names of all genres of songs performed by Keith Jarrett
   (e) the names of composers none of whose songs have been performed
   (f) the names of all entities (people or ensembles) involved with the song, "Anything Goes," either as composers or performers

   Answer: There are often more than one correct answer for each of these. Answers can also be in tree form or by naming intermediate subexpressions.

   (a) $\pi_{\texttt{songTitle}}(\sigma_{\texttt{composerName=Cole Porter}}(\texttt{Writes}))$
   (b) $\pi_{\texttt{composerName,songTitle}}(\sigma_{\texttt{composerName=perfName}}(\texttt{Writes} \bowtie \texttt{Performance}))$
   (c) $\pi_{\texttt{perfDate,perfName}}($
   $\sigma_{\texttt{songTitle=Anything Goes} \wedge \texttt{perfDate} \geq 1/1/2015 \wedge \texttt{perfDate} < 1/1/2016 \wedge \texttt{recorded=T}}(\texttt{Performance}))$
   (d) $\pi_{\texttt{name}}(\sigma_{\texttt{title=songTitle} \wedge \texttt{code=genreCode}}(\texttt{Genre} \times (\texttt{Song} \bowtie \sigma_{\texttt{perfName=Keith Jarrett}}(\texttt{Performance}))))$
   (e) $\pi_{\texttt{composerName}}(\texttt{Writes}) - \pi_{\texttt{composerName}}(\texttt{Writes} \bowtie \texttt{Performance})$
   (f) $\pi_{\texttt{composerName}}(\sigma_{\texttt{songTitle=Anything Goes}}(\texttt{Writes} \bowtie \texttt{Performance}))$
   $\cup \pi_{\texttt{perfName}}(\sigma_{\texttt{snogTitle=Anything Goes}}(\texttt{Performance}))$

2. (12 points) Write an SQL query (SELECT statement) that for each performed song returns the song title, average length of its performances, and its earliest performance date.[1]

   Answer:

```
SELECT songTitle, AVG(length), MIN(perfDate)
FROM Performance
GROUP BY songTitle;
```

3. (10 points) Write a simple, concise SQL query that returns the titles of those songs that have been performed at least 100 times.

   Answer:

[1] Dates can be compared chronologically using $<$, $\leq$, etc., where $d_1 < d_2$ means that date d_1 happens before date d_2. Any reasonable date format is OK.

```
SELECT songTitle
FROM Performance
GROUP BY songTitle
HAVING COUNT(perfDate) >= 100;
```

One could also use `COUNT(*)` for the count instead.

4. (20 points) Suppose the tables described above are made up of the following tuples, where for ease of reference, rows are sorted by the primary key (if there is one):

Genre: (sorted by code)

code	*name*
alt	Alternative
amb	Ambient
clas	Classical
cnty	Country
folk	Folk
hiho	Hip Hop
jazz	Jazz
rb	Rhythm & Blues
regg	Reggae

Song: (sorted by title)

title	*genreCode*	*cDate*
Anything Goes	jazz	4-JAN-1934
Auf dem Strom	clas	4-SEP-1828
I Shot the Sheriff	regg	23-MAY-1973
Psycho Killer	alt	2-DEC-1975
The Book I Read	alt	19-DEC-1976

Composer: (sorted by name)

name	*DOB*
Bob Marley	6-FEB-1945
Chris Frantz	8-MAY-1951
Cole Porter	9-JUN-1891
David Byrne	14-MAY-1952
Franz Schubert	31-JAN-1797
Tina Weymouth	22-NOV-1950

Writes: (sorted by song title)

composerName	*songTitle*
Cole Porter	Anything Goes
Franz Schubert	Auf dem Strom
Bob Marley	I Shot the Sheriff
Chris Frantz	Psycho Killer
David Byrne	Psycho Killer
Tina Weymouth	Psycho Killer
David Byrne	The Book I Read

Performance: (sorted by song title then performance date)

perfName	*songTitle*	*length*	*perfDate*	*recorded*
Ethel Merman	Anything Goes	195	21-NOV-1934	F
Frank Sinatra	Anything Goes	215	17-OCT-1955	T
Brad Mehldau Trio	Anything Goes	428	23-FEB-2004	T
Tony Bennett/Lady Gaga	Anything Goes	204	29-JUL-2014	T
Lewis/Brain/Lush	Auf dem Strom	584	6-APR-1954	T
Ainsley/Johnson/Schade	Auf dem Strom	569	1-JUN-1998	T
The Wailers	I Shot the Sheriff	237	30-NOV-1973	T
Eric Clapton	I Shot the Sheriff	230	4-JUL-1974	T
The Wailers	I Shot the Sheriff	318	17-JUL-1975	F
Talking Heads	Psycho Killer	259	28-DEC-1977	T
Talking Heads	Psycho Killer	259	19-AUG-1984	F
Talking Heads	The Book I Read	180	28-DEC-1977	T

What is returned by the following SQL queries? Give your answer in tabular form, including column headings. (You are free to suppress duplicate entries if you want.)

(a)
```
SELECT songTitle
FROM Performance
WHERE perfDate BETWEEN 1-JAN-1950 AND 31-DEC-1969;
```

(b)
```
SELECT composerName, length
FROM Song, Writes, Performance
WHERE title = Writes.songTitle AND title = Performance.songTitle AND
      genreCode = alt AND length >= 200;
```

(c)
```
SELECT perfName, MAX(length) maxLength
FROM Performance, Writes, Composer
WHERE Performance.songTitle = Writes.songTitle AND name = composerName AND
      DOB >= 1-JAN-1940
GROUP BY perfName;
```

Answer: Order of the rows does not matter. Duplicates can be suppressed in part (b).

(a)

songTitle
Anything Goes
Auf dem Strom

(b)

composerName	*length*
David Byrne	259
Chris Frantz	259
Tina Weymouth	259
David Byrne	259
Chris Frantz	259
Tina Weymouth	259

(c)

perfName	*maxLength*
Talking Heads	259

5. (25 points total) Let $R(A, B, C, D)$ satisfy the FDs $A \to D$, $AB \to C$, $C \to D$.

   (a) (5 points) List all the keys for R.

   (b) (10 points) List all BCNF violations. Give each violation in the form, $S^+ = T$, where S and T are sets of attributes, T contains an attribute not in S, but T does not contain all attributes.

   (c) (10 points) Use the information in the last item to decompose R fully into BCNF. Your decomposition should be efficient. Give your decomposition in tree form, as depicted in class. (There are two correct answers; you should only give one of them.)

Answer:

(a) The only key is AB.

(b) There are four BCNF violations:

$$A^+ = AD$$
$$C^+ = CD$$
$$AC^+ = ACD$$
$$BC^+ = BCD$$

(c) Here are the two efficient decompositions, the first using $A^+ = AD$ and the second using $C^+ = CD$:

$R(A, B, C, D)$
$R_1(A, D)$ $\qquad\qquad$ $R_2(A, B, C)$

$R(A, B, C, D)$
$R_1(C, D)$ $\qquad\qquad$ $R_2(A, B, C)$