

# CSE 414 Final Examination

June 6, 2016, 2:30pm - 4:20pm

Name: \_\_\_\_\_

Question	Points	Score
1	60	
2	30	
3	10	
4	50	
5	50	
Total:	200	

- This exam is CLOSED book and CLOSED devices.
- You are allowed TWO letter-size pages with notes (both sides).
- You have 1h:50 minutes; budget time carefully.
- Please read all questions carefully before answering them.
- Some questions are easier, others harder; if a question sounds hard, skip it and return later.
- Good luck!

# 1 SQL and Relational Languages

1. (60 points)

An online picture sharing company uses a database with the following schema:

Users(uid, uname, city)

Picture(pid, author, size, pdf)

- Users stores all users; uid is the key.
- Picture stores their pictures; pid is the key; author is the uid of the picture's author; size represents the size of the picture in bytes; pdf is the actual pdf content of the picture.
- uid, pid, author, size are integers; uname, city, pdf are text.

Users(uid, uname, city)  
Picture(pid, author, size, pdf)

- (a) (10 points) Write the SQL statements to create the tables for this database.

Users(uid, uname, city)  
Picture(pid, author, size, pdf)

- (b) (5 points) Write a SQL query that returns all users that have posted both a picture larger than 1MB (`size > 1000000`) and a picture smaller than 1MB. Your query should return the users' `uid` and `uname`

Users(uid, uname, city)  
Picture(pid, author, size, pdf)

- (c) (10 points) Write a SQL query that retrieves all users who do not have any picture greater than 1MB (`size > 1000000`). Your query should return the users' `uid` and `uname`

Users(uid, uname, city)  
Picture(pid, author, size, pdf)

- (d) (5 points) Write a Relational Algebra expression that is equivalent to the following SQL query:

```
select x.uid, x.uname, avg(y.size) as s
from Users x, Picture y
where x.uid = y.author
      and x.city = 'Seattle'
      and y.size > 20000
group by x.uid, x.uname
having count(*) > 10;
```

You may either write a Relational Algebra expression, or draw a query plan.

Users(uid, uname, city)  
 Picture(pid, author, size, pdf)

(e) Consider the following query:

```
select x.uid, x.uname,
       (select count(*)
        from Picture y
        where x.uid = y.author and y.size > 1000000)
from Users x
where x.city = 'Denver';
```

For each query below indicate whether it is equivalent to the given query (meaning: it returns the same answers). All queries are syntactically correct.

i. (2 points) Q1:

```
select x.uid, x.uname, count(*)
from Users x, Picture y
where x.uid = y.author and x.city = 'Denver' and y.size > 1000000
group by x.uid, x.uname;
```

i. \_\_\_\_\_

Your answer:

ii. (2 points) Q2:

```
select x.uid, x.uname, count(y.pid)
from Users x left outer join Picture y on x.uid = y.author and y.size > 1000000
where x.city = 'Denver'
group by x.uid, x.uname, x.city;
```

ii. \_\_\_\_\_

Your answer:

iii. (2 points) Q3:

```
select x.uid, x.uname, count(*)
from Users x left outer join Picture y on x.uid = y.author and y.size > 1000000
group by x.uid, x.uname, x.city
having x.city = 'Denver';
```

iii. \_\_\_\_\_

Your answer:

iv. (2 points) Q4:

```
select x.uid, x.uname, count(y.pid)
from Users x left outer join Picture y on x.uid = y.author and y.size > 1000000
group by x.uid, x.uname, x.city
having x.city = 'Denver';
```

iv. \_\_\_\_\_

Your answer:

v. (2 points) Q5:

```
select x.uid, x.uname, count(*)
from Users x, Picture y
where x.uid = y.author and y.size > 1000000 and x.city = 'Denver'
group by x.uid, x.uname;
```

v. \_\_\_\_\_

Your answer:

Users( <u>uid</u> , uname, city)		Artwork( <u>wid</u> , title, pdf)
Picture( <u>pid</u> , author, size, pdf)		Author( <u>aid</u> ,wid,aname)

- (f) The company merges with a local advertising company in Denver, which has a database of artistic pictures. Each artistic picture has a title, the pdf image, and may have several authors; all authors are from Denver.

```
drop table if exists Artwork;
drop table if exists Author;

create table Artwork (
  wid int primary key,
  title text,
  pdf text);

create table Author (
  aid int primary key,
  wid int not null references Artwork(wid),
  aname text not null);
```

- i. (10 points) Some users and pictures occur in both databases. Write a SQL query that finds all common user,picture pairs. Two users are considered to be the same if their names and cities are the same; two pictures are considered to be the same if their pdf's are the same. Your query should return **uname**, **uid**, **aid**, **pid**, **wid**: the user name (which is the same in both databases), its two keys in **Users** and **Author**, and the two keys of the identical picture in **Picture** and **Artwork**.

Users( <u>uid</u> , uname, city)		Artwork( <u>wid</u> , title, pdf)
Picture( <u>pid</u> , author, size, pdf)		Author( <u>aid</u> ,wid,aname)

- ii. (10 points) The new company creates a new schema for the integrated data:

```
create table NewUsers (  
    nuid int primary key,  
    nuName text not null,  
    city text not null);  
  
create table NewPicture (  
    npid int primary key,  
    title text,  
    size int,  
    pdf text);  
  
create table Authored (  
    nuid int not null references newUsers(nuid),  
    npid int not null references NewPicture(npid));
```

Write a sequence of SQL queries that inserts all the data from the two old databases into the new database. You do not need to eliminate duplicates: that is, you will insert all the records from `Users`, `Picture` into the new schema, then will insert all the records from `Artwork`, `Author` into the new schema, without worrying about duplicates. All keys in the `Users`, `Picture` database are distinct from the keys in the `Artwork`, `Author` database.

(This page is intentionally left blank.)

## 2 Database design

2. (30 points)

(a) (5 points) You are running a startup company allowing users to post blogs, and to comment on other users' blogs. Design the E/R diagram for your company's data. Your database should store information about users, bloggers, and blogs, and encode the following information:

- Every user has a `user ID` and a `name`.
- Every blog has a `blog ID`, a `text content` and one `author`, who is a blogger.
- Every blogger is a user.
- Every blogger also has a `rating` attribute.
- Users may comment on blogs.
- Every comment has an optional `text content`, and/or an optional `rating`.

Design the E/R diagram for this database.

(b) (10 points) Consider the following table:

FirstName	FirstInitial	LastName	LastInitial	FullName	FullAddress	ZipCode
"Alice"	"A"	"Levy"	"L"	"Alice Levy"	"45h St, Seattle, 98185",	"98185"
"Bob"	"B"	"Levy"	"L"	"Bob Levy"	"45h St, Seattle, 98185",	"98185"
"Alice"	"A"	"Davis"	"D"	"Alice Davis"	"Oak Ave, Seattle, 98185",	"98185"
"Carol"	"C"	"Davis"	"D"	"Carol Davis"	"Sunnyvale, 94085",	"94085"
"Carol"	"C"	"Louis"	"L"	"Carol Louis"	"Oak Ave, Seattle, 98185",	"98185"

Find all functional dependencies that hold on this table. You only need to write a minimal set of functional dependencies that logically imply all others.

- (c) (10 points) Using the functional dependencies you have identified at the previous question, decompose the relation in BCNF. Show your work, and the final normalized schema including all keys. Then represent the relation instance below in your normalized schema:

FirstName	FirstInitial	LastName	LastInitial	FullName	FullAddress	ZipCode
"Alice"	"A"	"Levy"	"L"	"Alice Levy"	"45h St, Seattle, 98185",	"98195"
"Bob"	"B"	"Levy"	"L"	"Bob Levy"	"45h St, Seattle, 98185",	"98195"
"Alice"	"A"	"Davis"	"D"	"Alice Davis"	"Oak Ave, Seattle, 98185",	"98195"
"Carol"	"C"	"Davis"	"D"	"Carol Davis"	"Sunnyvale, 94085",	"94085"
"Carol"	"C"	"Louis"	"L"	"Carol Louis"	"Oak Ave, Seattle, 98185",	"98195"

(d) (5 points) Consider the following database schema:

```
create table customer (  
    cid int primary key,  
    cname text,  
    city text);  
create table product (  
    pid int primary key,  
    pname text,  
    price int);  
create table orders (  
    cid int references customer,  
    pid int references product);
```

Consider the following query:

```
select x.cid, x.cname, x.city, z.pid, z.pname, z.price  
from customer x, orders y, product z  
where x.city = 'Seattle'  
    and x.cid = y.cid  
    and y.pid = z.pid  
    and z.price = 100;
```

Write all functional dependencies that hold on the query answer.

### 3 Semistructured Data and JSon

3. (10 points)

(a) For each small data instance below indicate whether we should call it relational data, or semi-structured data:

i. (1 point) Is this data:

name	email	age
Alice	a@li.ce	31
Bob	b@ob.com	24
Carol	NULL	37
David	da@vid.el	NULL
Erol	e@rol	22

i. \_\_\_\_\_

relational or semi-structured?

ii. (1 point) Is this data:

```

name,email,age
Alice,a@li.ce,31
Bob,b@ob.com,24
Carol,,37
David,da@vid.el,
Erol,e@rol,22

```

ii. \_\_\_\_\_

relational or semi-structured?

iii. (1 point) Is this data:

```
{ "Person"
  [ {"name": "Alice", "email": "a@li.ce", "age": 31},
    {"name": "Bob", "email": "b@ob.com", "age": 24},
    {"name": "Carol", "age": 37},
    {"name": "David", "email": "da@vid.el"},
    {"name": "Erol", "email": "e@rol", "age": 22}
  ]
}
```

iii. \_\_\_\_\_

relational or semi-structured?

iv. (1 point) Is this data:

```
{ "Person"
  [ {"name": "Alice", "email": ["a@li.ce", "alice@gmail.com"], "age": 31},
    {"name": "Bob", "email": "b@ob.com", "age": 24},
    {"name": "Carol", "age": 37},
    {"name": {"first": "David", "last": "Brown"}, "email": "da@vid.el"},
    {"name": "Erol", "email": "e@rol", "age": 22}
  ]
}
```

iv. \_\_\_\_\_

Relational or semi-structured?

(b) Consider the following applications. For each, indicate whether you would use a relational data model, or a semistructured data model:

- i. (3 points) You work for a large, national financial institution. Every night, all databases from the local branches of your institution sent over the internet all their updates for the day to a central server. The updates are complex in structure and may involve accounts, users, financial transactions, etc. What data model would you use to model the data exchange?

i. \_\_\_\_\_

Semi-structured or relational?

- ii. (3 points) You work for a large online shopping company, and maintain their Orders database. They have tens of millions of orders, and need to access about 2-300 orders per second; orders are retrieved either by the order ID, or by the date, or by the customer. What data model would you use for the orders database?

ii. \_\_\_\_\_

Semi-structured or relational?

## 4 Transactions

4. (50 points)

(a) For each statement below, indicate whether it is true or false:

- i. (3 points) In a static database, every serializable schedule is also conflict-serializable.

i. \_\_\_\_\_

Answer Yes/No:

- ii. (3 points) In a static database, every conflict-serializable schedule is also serializable.

ii. \_\_\_\_\_

Answer Yes/No:

- iii. (3 points) SQL Lite uses optimistic concurrency control.

iii. \_\_\_\_\_

Answer Yes/No:

- iv. (3 points) In a static database, strict Two-Phase-Locking is guaranteed to produce a serializable schedule.

iv. \_\_\_\_\_

Answer Yes/No:

- v. (3 points) In a static database, strict Two-Phase-Locking is guaranteed to produce a conflict-serializable schedule.

v. \_\_\_\_\_

Answer Yes/No:

- vi. (3 points) In a static database, strict Two-Phase-Locking is guaranteed to avoid deadlocks.

vi. \_\_\_\_\_

Answer Yes/No:

(b) For each of the schedules below, indicate whether they are conflict-serializable. If you answer *yes*, then give the equivalent serial order of the transactions. Show your work.

- i. (6 points) Is this schedule conflict-serializable? Show your work; if you answer 'yes', then indicate a serialization order.

R1(A), R1(B), W1(A), R2(B), W2(D), R3(C), R3(B), R3(D), W2(B), W1(C), W3(D)
---

- ii. (6 points) Is this schedule conflict-serializable? Show your work; if you answer 'yes', then indicate a serialization order.

R1(A), R1(B), W1(A), R2(B), W2(A), R3(C), R3(B), R3(D), W2(B), W1(C), W3(D)
---

- (c) A scheduler uses the strict two-phase locking protocol. In each of the cases below, indicate whether the scheduler may result in a deadlock. If you answer *yes*, then give an example of a schedule that results in deadlock.

- i. (5 points) Can these transactions result in deadlock?

T1:	W1(A), W1(C), CO1
T2:	W2(B), W2(D), CO2
T3:	W3(A), W3(B), CO3
T4:	W4(D), W4(A), CO4

Answer 'yes' or 'no'. If you answer 'yes' then also indicate a schedule that results in deadlock:

- ii. (5 points) Can these transactions result in deadlock?

T1:	W1(A), W1(C), CO1
T2:	W2(B), W2(D), CO2
T3:	W3(A), W3(B), CO3
T4:	W4(B), W4(C), CO4
T5:	W5(C), W5(D), CO5

Answer 'yes' or 'no'. If you answer 'yes' then also indicate a schedule that results in deadlock:

- (d) (10 points) We run the four transactions below concurrently on sqlite. Consider the following schedule:

Time	T1	T2	T3	T4
1	begin transaction			
2	select * from R where A = 1			
3		begin transaction		
4		select * from R where A = 2		
5	update R set B=10 where A=1			
6			begin transaction	
7			select * from R where A = 3	
8	commit			
9				begin transaction
10				select * from R where A = 4
11		commit		
12			commit	
13				commit

Circle the first action that will not be permitted by sqlite. Then modify the schedule, by filling out the table below, to reflect the actual schedule on sqlite. You can only delay actions, not move them earlier in time.

Time	T1	T2	T3	T4
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				

## 5 Parallel Data Processing

5. (50 points)

(a) We are running a MapReduce job over HDFS with a block size of 100KB. The input file has 1TB=  $10^{12}$  Bytes. Answer each of the questions below.

i. (3 points) How many map tasks will the MapReduce system create by default? If there is no default, then indicate so.

i. \_\_\_\_\_

Number of map tasks:

ii. (3 points) How many reduce tasks will the MapReduce system create by default? If there is no default, then indicate so.

ii. \_\_\_\_\_

Number of reduce tasks:

iii. We are running a MapReduce job over HDFS with a block size of 100KB. Most of the computation time is spend in the Map phase; the Reduce phase is very fast. Answer each question below:

$\alpha$ ) (4 points) Our input file has size 1TB=  $10^{12}$  Bytes. When we run the MapReduce job on 10 workers, the job takes 500 minutes. How long will the job take if we use 100 workers?

$\alpha$ ) \_\_\_\_\_

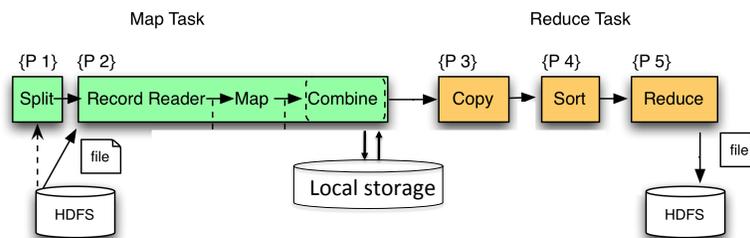
Write the number of minutes:

$\beta$ ) (4 points) Our input file has size 50KB=  $5 \cdot 10^4$  Bytes. When we run the MapReduce job on 10 workers, the job takes 10 minutes. How long will the job take if we use 100 workers?

$\beta$ ) \_\_\_\_\_

Write the number of minutes:

For the next few questions, recall the steps of a MapReduce job from the lecture notes:



- iv. (3 points) The *Copy* phase of the reduce tasks may start immediately after the first map tasks finish without having to wait for all map tasks to finish. (Recall that MapReduce implements the shuffle phase by having each reducer copy its values from the mappers.)

iv. \_\_\_\_\_

Yes or no?

- v. (3 points) The *Sort* phase of the reduce tasks may start immediately after the first map tasks finish without having to wait for all map tasks to finish.

v. \_\_\_\_\_

Yes or no?

- vi. (3 points) The *Reduce* phase of the reduce tasks may start immediately after the first map tasks finish without having to wait for all map tasks to finish.

vi. \_\_\_\_\_

Yes or no?

(b) The following questions compare MapReduce to Spark. For each statement indicate whether it is true or false.

- i. (3 points) In order to cope with worker failure, MapReduce stores all intermediate results to disk.

i. \_\_\_\_\_

True or False?

- ii. (3 points) In order to cope with worker failure, Spark stores all intermediate results to disk.

ii. \_\_\_\_\_

True or False?

- iii. (3 points) If a worker fails during the execution of a MapReduce program, then the entire program needs to be restarted.

iii. \_\_\_\_\_

True or False?

- iv. (3 points) If a worker fails during the execution of a Spark program, then the entire program needs to be restarted.

iv. \_\_\_\_\_

True or False?

- v. (3 points) Which of the following three Spark transformations corresponds to the `map` phase of a MapReduce program?
1. `map(f) : RDD[T] ⇒ RDD[U]`, where  $f : T ⇒ U$ .
  2. `flatMap(f) : RDD[T] ⇒ RDD[U]`, where  $f : T ⇒ Seq[U]$ .
  3. `map(f) : RDD[T] ⇒ RDD[T]`, where  $f : T ⇒ Bool$ .

v. \_\_\_\_\_

Answer 1,2, or 3:

- vi. What is the difference between `RDD[T]` and `Seq[T]`? Select all answers that apply.

$\alpha$ ) (3 points) An RDD can be longer than a Sequence.

 $\alpha$ ) \_\_\_\_\_

True or false?

$\beta$ ) (3 points) An RDD is distributed while a Sequence is stored on a single node.

 $\beta$ ) \_\_\_\_\_

True or false?

$\gamma$ ) (3 points) An RDD can be nested, e.g. we can have a data type `RDD[(K, RDD[T])]`.

 $\gamma$ ) \_\_\_\_\_

True or false?

$\delta$ ) (3 points) A Sequence can be nested, e.g. we can have a data type `Seq[(K, Seq[T])]`.

 $\delta$ ) \_\_\_\_\_

True or false?