Practice Final Exam Questions 2024

These are a few sample questions for the final exam. This is not representative of the length, distribution or coverage of questions, but is a sample of the type of questions that will be asked.

Question 1.
Indicate whether each of the following statements is TRUE or FALSE.

1. Using the relational algebra operators (set semantics) and relations $R(A, B), S(A, B)$
   $R \bowtie S = (R \cup S) \cap (S \cup R)$.

   True False

2. In materialized views, it is always possible to synchronize updates between the base table and the view.

   True False

3. In a 3NF decomposition, we are guaranteed dependency preservation and lossless joins.

   True False

4. Conflict serializable schedules are recoverable and avoid cascading aborts.

   True False

5. In the wound-wait policy, higher priority transactions always receive the requested lock(s) immediately.

   True False

6. In 2PL with lock conversions, a transaction can request a lock after downgrading only if the actions performed on the data object involve reads only.

   True False

Question 2.
Consider the two relational instances, *Emps* and *WorksIn*. Compute the result of the following query on these instances.

Emps:

Emp	Sal	Mgr
Xin	100	Pat
Jen	200	Jen
Pat	100	Meg
Meg	150	Meg
Joe	110	Ami
Ami	200	Bob

WorksIn:

Dept	Emp
Toy	Xin
Car	Bob
Car	Meg
Art	Pat
Art	Meg

```
SELECT *
FROM Emps,
    (select Emps.emp, max(sal) as sal
    from Emps, WorksIn
    where Emps.emp = WorksIn.emp
    group by Emps.emp) as foo
WHERE Emps.mgr = foo.emp
```

Question 3.

Consider the following schema.

– eid: unique employee id
– name: employee's name
– city: employee's city of residence
Employee(eid,name,city)

– cid: unique company id
– name: company name
– city: city where company is located
Company(cid,name,city)

– eid: foreign key of Employee
– cid: foreign key of Company
– an employee can work for a company at most once
– salary: employee's salary for the said company in CAD
WorksFor(eid,cid,salary)

– eid: foreign key of Employee
– cid: foreign key of Company
– mid: employee id of eid's manager (foreign key of Employee)
Manages(eid,cid,mid)

(a) Write an SQL query that returns the names of all managers, along with the highest number of employees they have managed in any single company (and the name of that company). For example, if manager Fred has managed two employees at IBM and three at Microsoft, then the query should return "(Fred, 3, Microsoft)". If there are ties (for example, Fred managed three people at IBM and three people at Microsoft), then a manager may be returned more than once (for this example, two tuples "(Fred, 3 Microsoft)" and "(Fred, 3, IBM)" would be returned).

Question 4.

Consider the following schema (keys are underlined):

Product(<u>pid</u>, name, type, mfgr, price), Buys(<u>cid</u>, <u>pid</u>), Customer(<u>cid</u>, cname, age, gender)

a) Write the following query in relational algebra: *Find the names of all customers who have purchased a product manufactured by 'Apple'.*

b) Write the following query in SQL: *Find the names of all customers who have not purchased the most expensive product.*

Question 5.

Consider the following schema for tracking customers' ratings of books for a bookstore website. Keys are underlined.

- Books(<u>ISBN</u>, title, author, year, length)
  ISBN is a string that is used internationally for identifying books.
- Authors(<u>AID</u>, name)
- Customers(<u>CID</u>, name)
- Ratings(<u>ISBN, CID</u>, rating)
  Rating is an integer.

The following foreign key (FK) constraints hold:

- Books(author) is FK to Authors(AID)
- Ratings(ISBN) is FK to Books(ISBN)
- Ratings(CID) is FK to Customers(CID)

Using the relational algebra operators $\Pi, \sigma, \bowtie, \times, \cap, \cup, -, \rho$, `:=`, write the following queries, assuming set semantics.

(a) Find the CID of every customer who has rated every book by author "Tolkien".

(b) Find the name of every author who has published exactly two books before the year 2000, and none since 2000 (*i.e.,* none in the year 2000 or later).

Question 6. Given the instance of four relations:

R:

A	B
1	2
3	4
1	3

S:

B	C
1	3
2	4

T:

B	C
2	5
2	3

V:

E	F
1	4
2	5

(a) What is the result of the following query (assuming set semantics)? Provide the resulting schema and result tuples.

$$\mathtt{V} \times (\pi_{\mathtt{B}}(\mathtt{R} \bowtie \mathtt{S}) - \pi_{\mathtt{B}}(\mathtt{T}))$$

(b) What is the result of the following query (assuming bag semantics)? Provide the resulting schema and tuples.

$$\pi_{\mathrm{A}}(\mathrm{R}) - \pi_{\mathrm{A}}(\rho_{\mathrm{A} \leftarrow \mathrm{B}}((\mathrm{S})))$$

Question 7. Illustrate with an example how views can be used for security. Your example should include a table, a view defined over the table, and a brief description of how the view hides some information from an unauthorized user.

Question 8.
Consider relation R(A,B,C,D) and functional dependencies

$$\{ABC \to D,\ CD \to A,\ CA \to B,\ AD \to C,\ CD \to B\}.$$

(a) List all the keys of R.

(b) Given a minimal cover of $\{AC \to D, AC \to B, CD \to A, AD \to C\}$, compute a 3NF decomposition of R.

(c) Is your decomposition dependency preserving?

(d) Consider relation R(A,B,C,D) and functional dependencies $\{D \to B,\ C \to A,\ A \to B\}$.

Is the decomposition of R into R1(A,B), R2(A,C) and R3(A,D) a lossless join decomposition? If not, give an example instance of R where the join of R1, R2, and R3 is not R.

(e) Is the decomposition of R into R1(A,B), R2(A,C) and R3(A,D) dependency preserving?

Question 9. Consider the relation R with schema R(A,B,C,D), and functional dependencies $\{AB \rightarrow C, C \rightarrow D, D \rightarrow A\}$.

(a) Is the relation in BCNF? Is it in 3NF? Explain why or why not.

(b) Is the decomposition of R into AB, BC and CD lossless? Why or why not?

(c) Is the decomposition of R into AB, BC and CD dependency preserving? Why or why not?

Question 10.

Prove the following inference rule for functional dependencies using only Armstrong's axioms:
If $P \to QR$ and $R \to S$, then $P \to QS$.

Show the steps of your proof, and indicate which of Armstrong's axioms is applied in each step.

Question 11. Indicate whether each of the following schedules is *conflict serializable*, *recoverable*, *avoids cascading aborts*, *2PL*, or *strict 2PL*.

a) $R_1(A), W_2(A), R_1(B), Commit_1, W_3(B), R_3(B), W_3(A), Commit_3, R_2(C), Commit_2$.

b) $R_1(A), W_2(B), R_1(B), Commit_1, Commit_2$

c) $R_1(A), W_2(B), R_1(B), Commit_2, Commit_1$

Question 12. [Pending, based on lecture material coverage.]

For each of the schedules below, describe how the given sequence of lock requests is handled. Assume that the timestamp of transaction Ti is i (so Ti has higher priority than Tj if $i < j$). If a transaction is blocked, assume that all of its actions are queued until it is resumed. Your answer should clearly indicate the **ordered** sequence of lock requests.

Consider the following sequence of actions, listed in the order they are submitted. Indicate the scheduled sequence of lock requests (assuming only shared and exclusive locks), using deadlock prevention based on the **Wound-Wait** policy. For each action, indicate the transaction, the type of lock requested, on which data object (X or Y), and any transaction that is blocked (forced to wait) or aborted.

For example, for schedule $W_2(Z), W_1(Z)$, you would state:

i) *Transaction T_2 is requesting an exclusive lock on object Z, no transaction is blocked or aborted.*

ii) *Transaction T_1 is then requesting an exclusive lock on object Z, and T_2 is aborted (using the Wound-Wait policy).*

(a) $R_1(X), W_2(Y), W_2(X), W_3(Y), W_1(Y)$

(b) Given the following sequence of actions, under a strict 2PL protocol with *deadlock detection*. Is there a deadlock? If so, show the waits-for graph.

$R_1(X), W_2(Y), W_2(X), W_3(Y), W_1(Y), Commit_1, Commit_2, Commit_3$