MCQs on Concurrency and Transaction Management in DBMS

Multiple Choice Questions

Q1. Consider the following log sequence of two transactions on a bank account, with initial balance 12000, that transfer 2000 to a mortgage payment and then apply a 5% interest.

1. T1 start
2. T1 B old=12000 new=10000
3. T1 M old=0 new=2000
4. T1 commit
5. T2 start
6. T2 B old=10000 new=10500
7. T2 commit (CRASH happens before this step)

When the system is restarted, which one statement is true of the recovery procedure?

A. We must redo log record 6 to set B to 10500

B. We must undo log record 6 to set B to 10000 and then redo log records 2 and 3.

C. We need not redo log records 2 and 3 because transaction T1 has committed.

D. We can apply redo and undo operations in arbitrary order because they are idempotent.

Answer: B

Q2. Consider the given schedule and choose the suitable option.
S = T1:R(x), T1:R(y), T1:W(x), T2:R(y), T3:W(y), T1:W(x), T2:R(y)

A. Schedule is view serializable

B. Schedule is conflict serializable but not view serializable

C. Schedule is view serializable but not conflict serializable

D. Neither view serializable nor conflict serializable

Answer: C

Q3. Which of the following scenarios may lead to an irrecoverable error in a database system?

A. A transaction writes a data item after it is read by an uncommitted transaction

B. A transaction reads a data item after it is read by an uncommitted transaction

C. A transaction reads a data item after it is written by a committed transaction

D. A transaction reads a data item after it is written by an uncommitted transaction

Answer: D

Q4. Consider the following transaction involving two bank accounts x and y:
`read(x); x := x - 50; write(x); read(y); y := y + 50; write(y);`
The constraint that the sum of the accounts x and y should remain constant is that of:

A. Atomicity

B. Consistency

C. Isolation

D. Durability

Answer: B

Q5. Consider a simple checkpointing protocol and the following set of operations in the log:

> (start, T4); (write, T4, y, 2, 3); (start, T1); (commit, T4); (write, T1, z, 5, 7);
> (checkpoint);
> (start, T2); (write, T2, x, 1, 9); (commit, T2); (start, T3); (write, T3, z, 7, 2);

If a crash happens now, what are the contents of the undo list and the redo list?

A. Undo: T3, T1; Redo: T2

B. Undo: T3, T1; Redo: T2, T4

C. Undo: none; Redo: T2, T4, T3, T1

D. Undo: T3, T1, T4; Redo: T2

Answer: A

Q6. Which level of locking provides the highest degree of concurrency in a relational database?

A. Page

B. Table

C. Row

D. Page, table and row level locking allow the same degree of concurrency

Answer: C

Q7. Consider the following schedule S of transactions T1 and T2:

T1	**T2**
Read(A); A = A - 10;	Read(A); Temp = 0.2 * A; Read(B);
Write(A); Read(B); B = B + 10; Write(B);	B = B + Temp; Write(B);

Which one of the following is TRUE?

A. S is serializable only as T1, T2

B. S is serializable only as T2, T1

C. S is serializable both as T1, T2 and T2, T1

D. S is serializable either as T1 or as T2

E. None of these

Answer: E

Q8. Consider the following two-phase locking protocol. Suppose a transaction T accesses (for read or write operations) a certain set of objects $\{O_1, \ldots, O_k\}$ in the following manner:

1. Step 1: T acquires exclusive locks to $O_1, \ldots, O_k$ in increasing order of their addresses
2. Step 2: The required operations are performed
3. Step 3: All locks are released

This protocol will:

A. Guarantee serializability and deadlock-freedom

B. Guarantee neither serializability nor deadlock-freedom

C. Guarantee serializability but not deadlock-freedom

D. Guarantee deadlock-freedom but not serializability

Answer: A

Q9. Suppose a database schedule S involves transactions $T_1, \ldots, T_n$. Construct the precedence graph of S with vertices representing the transactions and edges representing the conflicts. If S is serializable, which one of the following orderings of the vertices of the precedence graph is guaranteed to yield a serial schedule?

A. Topological order

B. Depth-first order

C. Breadth-first order

D. Ascending order of transaction indices

Answer: A

Q10. Consider the following database schedule with two transactions, T1 and T2.
S = r2(X); r1(X); r2(Y); w1(X); r1(Y); w2(X); a1; a2;

A. S is non-recoverable

B. S is recoverable, but has a cascading abort

C. S does not have a cascading abort

D. S is strict

Answer: C