USN

Internal Assessment Test 3– August 2024

Sub:	Database Management System					Sub Code:	BCS403	CSE	
Date:	5.7.2024	Duration:	90 mins	Max Marks:	50	Sem / Sec:	IV/ A, B, C	OBE	

Answer any FIVE FULL Questions

		MARK	CO	RBT
1	Why concurreny control and recovery are needed in DBMS? Explain the different problems occur in concurrent transactions?	10	CO 4	L2,
2a 2b	Explain ACID properties of Transaction? Explain the state transition diagram of a transaction?	5+5	CO 4	L2
3	Breifly explain 2PL(2 phase locking) protocol for concurrency control.?How does it guarantee serialisability? Explain with example?	10	CO 5	L2

USN

Internal Assessment Test3 – August 2024

Sub:	Database Management System					Sub Code:	BCS403	CSE	
Date:	5.7.2024	Duration:	90 mins	Max Marks:	50	Sem / Sec:	IV/ A, B, C	OBE	

Answer any FIVE FULL Questions

		MARKS	CO	RBT
1	Why concurreny control and recovery are needed in DBMS? Explain the different problems occur in concurrent transactions?	10	CO 4	L2,
2a 2b	Explain ACID properties of Transaction? Explain the state transition diagram of a transaction?	5+5	CO 4	L2
3	Breifly explain 2PL(2 phase locking) protocol for concurrency control.?How does it guarantee serialisability?Explain with example?	10	CO 5	L2

4	Write short notes on a)cascading rollback and transaction rollback b) Transaction support in sql	5+5	C04	L2
5	Describe the characteristics of NOSQL system?Explain CAP theorem?	[6+4]	CO6	L2
6	Consider the three transactions T1,T2,T3 and schedules S1 and S2 given below.Draw the serialisability (precedence) graphs for S1 and S2 and state whether each schedule is serialisable or not.If a schedule is serialiazable, write down the equivalent serial schedule(s). T1:r1(X);r1(Z);w1(X); T2:r2(Z);r2(Y);w2(Z);w2(Y); T3:r3(X);r3(Y);w3(Y); S1:r1(X),r2(Z);r1(Z);r3(X);r3(Y);w1(X);w3(Y);r2(Y);w2(Z);w2(Y); S2:r1(X);r2(Z);r3(X);r1(Z);r2(Y);r3(Y);w1(X);w2(Z);w3(Y);w2(Y);	10	CO4	L3

4	Write short notes on a)cascading rollback and transaction rollback b) Transaction support in sql	5+5	C04	L2
5	Describe the characteristics of NOSQL system? Explain CAP theorem?	[5+5]	CO6	L2
6	Consider the three transactions T1,T2,T3 and schedules S1 and S2 given below.Draw the serialisability (precedence) graphs for S1 and S2 and state whether each schedule is serialisable or not.If a schedule is serialiazable, write down the equivalent serial schedule(s). T1:r1(X);r1(Z);w1(X); T2:r2(Z);r2(Y);w2(Z);w2(Y); T3:r3(X);r3(Y);w3(Y); S1:r1(X),r2(Z);r1(Z);r3(X);r3(Y);w1(X);w3(Y);r2(Y);w2(Z);w2(Y); S2:r1(X);r2(Z);r3(X);r1(Z);r2(Y);r3(Y);w1(X);w2(Z);w3(Y);w2(Y);	10	CO4	L3

IAT3 -ANSWERKEY

Answer1:

Concurrency control is a very important concept of DBMS which ensures the simultaneous execution or manipulation of data by several processes or users without resulting in data inconsistency. Concurrency control provides a procedure that is able to control concurrent execution of the operations in the database.
recovery-maintaining the consistency and reliability of data.

The problems of concurrency are,

Lost Update Problem(write-write conflict)

Temporary Update Problem(dirty read problem

Incorrect Summary Problem

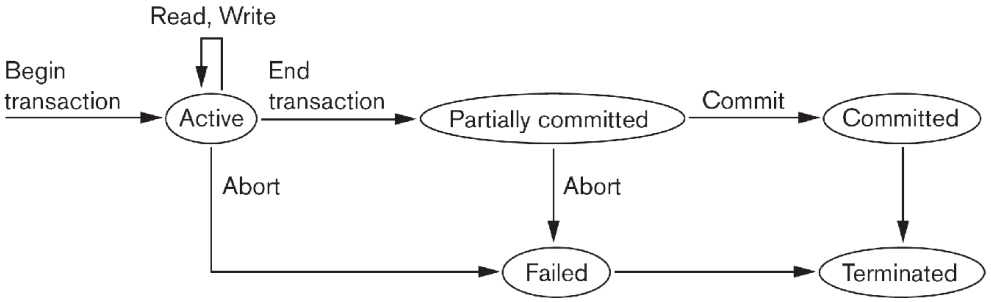
Unrepeatable read problem

Answer 2.
ACID properties

—-Atomicity

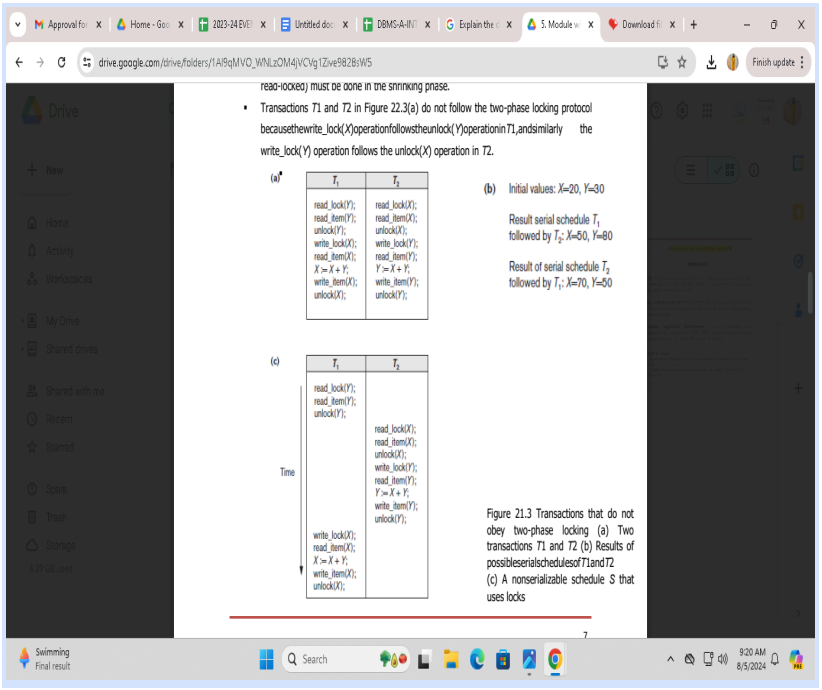
Consistency
Integrity
Durability

Figure 21.4
State transition diagram illustrating the states for transaction execution.



Answer 3

A transaction is said to follow the two-phase locking protocol if all locking operations
(read_lock, write_lock) precede the first unlock operation in the transaction
Such A Transaction Can Be Divided Into Two Phases:
Expanding Or Growing(first)phase,during which locks items can be acquired but none can be released
Shrinking(second)phase, during which existing locks can be released but no new
locks can be acquired
Iflockconversionisallowed,thenupgradingoflocks(from read-locked write-locked)
mustbedoneduringthe expanding phase,and downgrading locks (fromwrite-locked to read-locked) must be done in the shrinking phase.


read-locked) must be done in the shrinking phase.

- Transactions T1 and T2 in Figure 22.3(a) do not follow the two-phase locking protocol because the write_lock(X) operation follows the unlock(Y) operation in T1, and similarly the write_lock(Y) operation follows the unlock(X) operation in T2.

(a)

T_1	T_2

read_lock(Y);
read_item(Y);
unlock(Y);
write_lock(X);
read_item(X);
X := X + Y;
write_item(X);
unlock(X);
	read_lock(X);
	read_item(X);
	unlock(X);
	write_lock(Y);
	read_item(Y);
	Y := X + Y;
	write_item(Y);
	unlock(Y);

(b) Initial values: X=20, Y=30

Result serial schedule T_1 followed by T_2: X=50, Y=80

Result of serial schedule T_2 followed by T_1: X=70, Y=50

(c)

T_1	T_2

read_lock(Y);
read_item(Y);
unlock(Y);	

	read_lock(X);
	read_item(X);
	unlock(X);
	write_lock(Y);
	read_item(Y);
	Y := X + Y;
	write_item(Y);
	unlock(Y);

write_lock(X);
read_item(X);
X := X + Y;
write_item(X);
unlock(X);	

Time

Figure 21.3 Transactions that do not obey two-phase locking (a) Two transactions T1 and T2 (b) Results of possible serial schedules of T1 and T2 (c) A nonserializable schedule S that uses locks

7

Answer 4.

Cascading rollback: Uncommitted transaction has to be rolled back because it reads an item from a transaction that failed.

Schedule S: r1(X); w1(X); r2(X); r1(Y); w2(X); a1; a2;

A cascading rollback occurs in database systems when a transaction (T1) causes a failure and a rollback must be performed. Other transactions dependent on T1's actions must also be rollback due to T1's failure, thus causing a cascading effect. That is, one transaction's failure causes many to fail.

You can use ROLLBACK TRANSACTION to erase all data modifications made from the start of the transaction or to a savepoint. It also frees resources held by the transaction. Rolling back a transaction doesn't include changes made to local variables or table variables. These changes aren't erased by this statement.

Transaction Support in SQL

- **Phantoms**: New row inserted after another transaction accessing that row was started.

  A transaction T1 may read a set of rows from a table (say EMP), based on some condition specified in the SQL WHERE clause (say DNO=5). Suppose a transaction T2 inserts a new EMP row whose DNO value is

5. T1 should see the new row (if equivalent serial order is T2; T1) or not see it (if T1; T2). The record that did not exist when T1 started is called a **phantom record**.

Sample SQL transaction:

```
EXEC SQL whenever sqlerror go to UNDO;
EXEC SQL SET TRANSACTION
         READ WRITE
         DIAGNOSTICS SIZE 5
         ISOLATION LEVEL SERIALIZABLE;
EXEC SQL INSERT
         INTO EMPLOYEE (FNAME, LNAME, SSN, DNO, SALARY)
         VALUES ('Robert','Smith','991004321',2,35000);
EXEC SQL UPDATE EMPLOYEE
         SET SALARY = SALARY * 1.1
         WHERE DNO = 2;
EXEC SQL COMMIT;
GO TO  THE_END;
UNDO: EXEC SQL ROLLBACK;
THE_END:  ...
```

Answer 5.

The key features of NoSQL include
-simple design, s
seamless horizontal scalability
, and granular availability control

CAP theorem or Eric Brewer's theorem states that we can only achieve at most two out of three guarantees for a database: Consistency, Availability, and Partition Tolerance. Consistency means that all nodes in the network see the same data at the same time.

The CAP theorem states that it is not possible to guarantee all three of the desirable properties – consistency, availability, and partition tolerance at the same time in a distributed system with data replication.

The theorem states that networked shared-data systems can only strongly support two of the following three properties:

- **Consistency –**

  Consistency means that the nodes will have the same copies of a replicated data item visible for various transactions. A guarantee that every node in a distributed cluster returns the same, most recent and a successful write. Consistency refers to every client having the same view of the data. There are various types of consistency models. Consistency in CAP refers to sequential consistency, a very strong form of consistency.

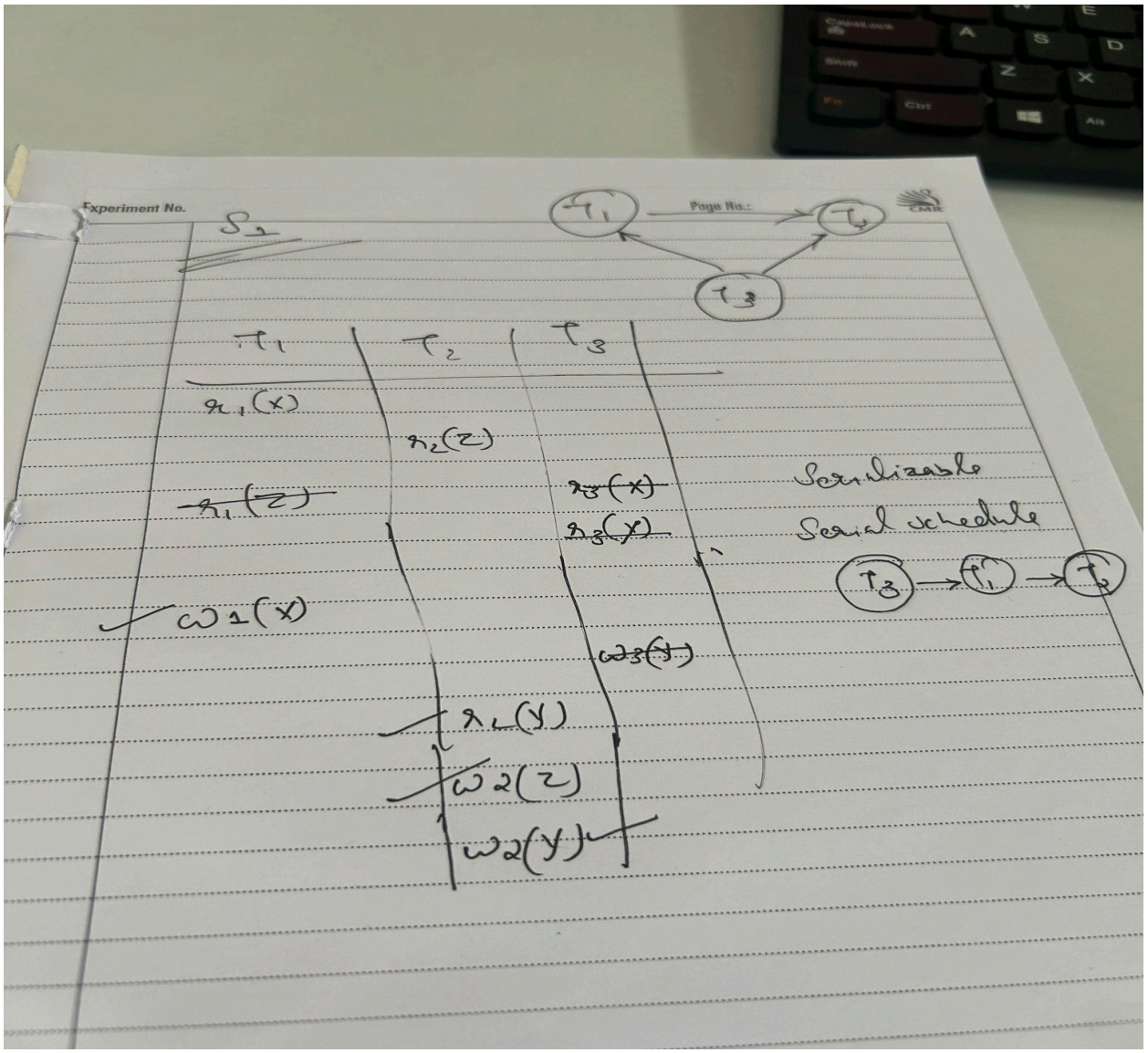
Availability –

Availability means that each read or write request for a data item will either be processed successfully or will receive a message that the operation cannot be completed. Every non-failing node returns a response for all the read and write requests in a reasonable amount of time. The key word here is “every”. In simple terms, every node (on either side of a network partition) must be able to respond in a reasonable amount of time.

Partition Tolerance –

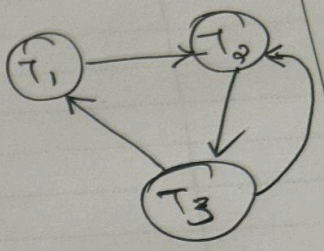
Partition tolerance means that the system can continue operating even if the network connecting the nodes has a fault that results in two or more partitions, where the nodes in each partition can only communicate among each other. That means, the system continues to function and upholds its consistency guarantees in spite of network partitions. Network partitions are a fact of life. Distributed systems guaranteeing partition .

Answer6


Experiment No.
Page No.:
T_1
T_2
T_3
$r_1(x)$
$r_2(z)$
$r_3(x)$
$r_1(z)$
$r_3(y)$
$w_1(x)$
$w_3(y)$
$r_2(y)$
$w_2(z)$
$w_2(y)$
Serializable
Serial schedule
$T_3 \rightarrow T_1 \rightarrow T_2$

S1

T_1	T_2	T_3
$r_1(X)$		
	$r_2(Z)$	
		~~$r_3(X)$~~
~~$r_1(Z)$~~		
	~~$r_2(Y)$~~	
		~~$r_3(Y)$~~
$w1(X)$		
	$w2(Z)$	
		$w_3(Y)$
	$w_2(Y)$	



non-conflict serializable.