Birla Institute of Technology & Science, Pilani

Work Integrated Learning Programmes Division

Final-Semester Test (Regular)

Course Number	:	SEWI Zxxxx
Course Title	:	< DBMS>
Type of Exam	:	Closed Book
Weightage	:	60 %
Duration	:	180 minutes
Date of Exam	:	DAY, dd-mm-yyyy

No. of Pages :
No. of Questions :

Session : FN
AN

Note:
1. Please read and follow all the instructions given on the cover page of the answerscript.
2. Start each answer from a fresh page. All parts of a question should be answered consecutively.

Q.1 Define the following terms **(2X5=10 Marks)**

- ✓ **Inner Join**
- ✓ **Locking**
- ✓ **NoSQL**
- ✓ **Optimizer**
- ✓ **Page**

Q.2 Consider the following relational schema and translate the following SQL-query into an expression of the relational algebra. **(5X2=10 Marks)**

- **Student** (snum, sname, major, level, age)
- **Class** (name, meets at, room, fid)
- **Enrolled** (snum, cname)
- **Faculty** (fid, fname, deptid)

Q2a)
```
SELECT S.sname
FROM Student S
WHERE S.snum
NOT IN (
        SELECT E.snum
        FROM
        Enrolled E)
```

Q2b)
```
SELECT C.name
FROM Class C
WHERE C.room = 'R128' OR C.name IN
        (SELECT E.cname
        FROM Enrolled E
        GROUP BY E.cname
        HAVING COUNT(*) >= 5)
```

Q.3a Normalize the table to 1NF **(3 Marks)**

CustomerName	Last4DigitsOfCard	CardType
Adam	3424 7632	CreditCard DebitCard
Alex	1413	CreditCard
Stuart	4721	DebitCard

Q.3b Normalize the table to 3NF **(4 Marks)**

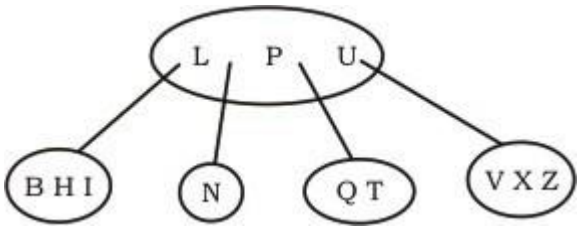
CustomerId	CustomerName	DOB	Street	City	State	Zip

Q.3c Normalize the table to 5NF **(4 Marks)**

AGENT	COMPANY	PRODUCT
Smith	Ford	car
Smith	Ford	truck
Smith	GM	truck
Smith	GM	car
Jones	Ford	truck

Q.4a) A FAT (file allocation table) based file system is being used and the total overhead of each entry in the FAT is 4 bytes in size. Given a 100 x 10^6 bytes disk on which the file system is stored and data block size is 10^3 bytes. What is the maximum size of a file that can be stored on this disk in units of 10^6bytes? **(5 Marks)**

Q4b) Consider the following 2-3-4 tree (i.e., B-tree with a minimum degree of two) in which each data item is a letter. The usual alphabetical ordering of letters is used in constructing the tree. What is the result of inserting G in the above tree? **(5 Marks)**



Q.5a) Consider the following partial Schedule S involving two transactions T1 and T2. Only the read and the write operations have been shown. The read operation on data item P is denoted by read (P) and the write operation on data item P is denoted by write (P). What would the consequence when the transaction T1 fails immediately after time instance 9. **(5 Marks)**

Time	Transaction-id	
	T1	*T2*
1	*read(A)*	
2	*write(A)*	
3		*read(C)*
4		*write(C)*
5		*read(B)*
6		*write(B)*
7		*read(A)*
8		*commit*
9	*read(B)*	

Q.5b) Consider two transactions: **(5 Marks)**

T1: BEGIN A=A+100, B=B-100 END

T2: BEGIN A=1.06*A, B=1.06*B END

- 1st TXN transfers $100 from B's account to A's
- 2nd TXN credits both accounts with 6% interest.

Assume at first A and B each have $1000. What are the legal outcomes of running T1 and T2???

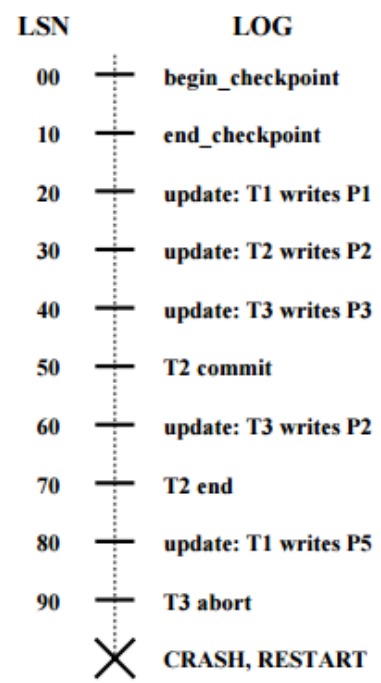
Q.6 **(2+4+4=10 Marks)**

Consider the execution shown in Figure In addition, the crashes system during recovery after writing two log records to stable storage and again after writing another two log records.

1. What is the value of the LSN stored in the master log record?

2. What is done during Analysis?

3. What is done during Redo?



Answers

Q.1 Define the following **(2X5=10 Marks)**

- ✓ **Inner Join** – A join between two tables where only the rows with matching foreign and primary key values are returned
- ✓ **Locking** – A method for safely protecting objects from being changed by two or more users (processes/threads) at the same time. A write (exclusive) lock allows access from only one user (process/thread) at a time. A read (shared) lock allows read-only access from multiple users (processes/threads).
- ✓ **NoSQL** – A classification of data storage systems which are not primarily designed to be relationally accessed through the common SQL language. NoSQL systems are characterized by dynamic creation and deletion of key/value pairs, and are structured to be highly scalable to multiple computers.
- ✓ **Optimizer** – A component of the SQL system that estimates the optimum, (i.e., fastest) method to access the database data that requested is by particular SQL SELECT, UPDATE, or DELETE statement.
- ✓ **Page** – The basic unit of database file input/output. Database files may b organized into a set of fixed-sized pages containing data associated with one or more record occurrences (table rows).

Q.2 Relational Algebra for the given SQL

Q2a)

$$\pi_{\texttt{S.sname}}(\rho_{\texttt{S}}(\texttt{Student}) \ \overline{\bowtie}$$
$$\pi_{\texttt{S.snum,S.sname,S.major,S.level,S.age}}\ \sigma_{\texttt{E.snum=S.snum}}\ (\rho_{\texttt{E}}(\texttt{Enrolled}) \times \rho_{\texttt{S}}(\texttt{Student})))$$

Q2b)

$$\pi_{\texttt{C.name}}\ \sigma_{\texttt{C.room='R128'}}\ \rho_{\texttt{C}}(\texttt{Class})$$
$$\cup\ \pi_{\texttt{C.cname}}\ \sigma_{\texttt{COUNT(*)>=5}}\ \gamma_{\texttt{E.cname,COUNT(*),C.name,C.meets\_at,C.room,C.fid}}$$
$$\sigma_{\texttt{E.cname=C.cname}}(\rho_{\texttt{E}}(\texttt{Enrolled}) \times \rho_{\texttt{C}}(\texttt{Class})).$$

Q.3a Normalize the table to 1NF **(3 Marks)**

CustomerId	CustomerName	Last4DigitsOfCard	CardType
101	Adam	3424	CreditCard
102	Adam	7632	DebitCard
103	Alex	1413	CreditCard
104	Stuart	4721	DebitCard

Q.3b Normalize the table to 3NF **(4 Marks)**

Note: ZipID is made as Primary Key in CustomerAddressTable and referred same as foreign key in CustomerTable. This table can also be normalized by doing CityID as Primary Key in CustomerAddressTable and referred same as foreign key in CustomerTable.

CustomerTable

CustomerID	CustomerName	DOB	ZipID

CustomerAddressTable

ZipID	Street	City	State

Q.3c

AGENT	COMPANY
Smith	Ford
Smith	GM
Jones	Ford

COMPANY	PRODUCT
Ford	car
Ford	truck
GM	car
GM	truck

AGENT	PRODUCT
Smith	car
Smith	truck
Jones	car

Q.4 a) **(5 Marks)**

Here block size is 10^3 B.

No. of entries in the FAT = Disk capacity/ Block size

= 10^8/10^3 = 10^5

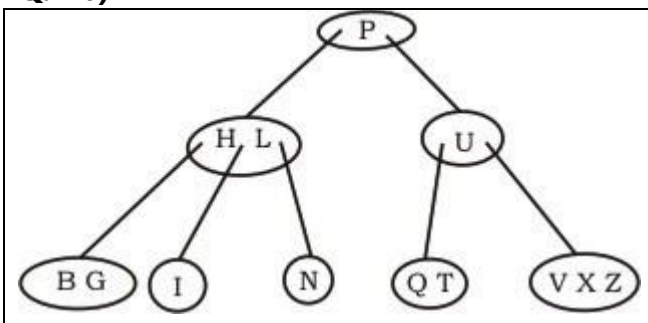
Total space consumed by FAT = 10^5 *4B = 0.4*10^6B

Max. size of file that can be stored = 100*10^6-0.4*10^6

= 99.6*10^6B.

So answer 99.6.

Q.4 b) **(5 Marks)**



Explanation: Since the given B tree is 2-3-4 tree, there can be at-most 4 children or 3 keys. In B Tree insertion, we start from root and traverse till the leaf node where key is to be inserted. While traversing, if we find a node which full, we split it. When we insert G, we find root itself is full, so we split it. When we come down to leftmost leaf, we find that the leaf is also full, so we split the leaf also.

Q.5a) Schedule S is non-recoverable and cannot ensure transaction atomicity because T2 reads value of 'A' which is written by T1 and T2 is committed before T1. **(5 Marks)**

Q.5b) There is no guarantee that T1 will execute before T2 or vice-versa, if both are submitted together. But, the net effect *must* be equivalent to these two transactions running serially in some order.

Legal outcomes: **A=1166, B=954 or A=1160, B=960** **(5 Marks)**

Q.6

1). LSN 00 is stored in master log record. It is the LSN of the begin checkpoint record.

2). At the end of analysis, the transaction table contains the following entries: (T1,80), and (T3,60). The Dirty Page Table has the following entries: (P1,20), (P2,30), (P3,40), and (P5,80)

LSN 20	Add (T1,20) to TT and (P1,20) to DPT
LSN 30	Add (T2,30) to TT and (P2,30) to DPT
LSN 40	Add (T3,40) to TT and (P3,40) to DPT
LSN 50	Change status of T2 to C
LSN 60	Change (T3,40) to (T3,60)
LSN 70	Remove T2 from TT
LSN 80	Change (T1,20) to (T1,70) and add (P5,70) to DPT
LSN 90	No action

3). Redo starts from LSN20 (minimum recLSN in DPT).

LSN 20	Check whether P1 has pageLSN more than 10 or not. Since it is a committed transaction, we probably need not redo this update
LSN 30	Redo the change in P2
LSN 40	Redo the change in P3
LSN 50	No action
LSN 60	Redo the changes on P2
LSN 70	No action
LSN 80	Redo the changes on P5
LSN 90	No action