**Birla Institute of Technology & Science, Pilani**

**Work-Integrated Learning Programmes Division**

**First Semester 2023-2024**

**Comprehensive Examination**

**(EC-3 Regular)**

Course No. : CSI ZG518

Course Title : Database Design and Applications

Pattern of Exam : TYPED ONLY

Nature of Exam : Open Book

No. of Pages = 8

# No. of Questions = 4

Weightage : 40%

Duration : 2 ½ Hours

Date of Exam : 26/11/2023 (AN)

Note:

1. Please follow all the *Instructions to Candidates* given on the cover page of the answer book.
2. All parts of a question should be answered consecutively. Each answer should start from a fresh page.
3. Assumptions made if any, should be stated clearly at the beginning of your answer.

(i) Consider the following classes of schedules: *serializable, conflict-serializable, view-serializable, recoverable, avoids-cascading-aborts,* and *strict*. For each of the following schedules, state which of the preceding classes it belongs to. If you cannot decide whether a schedule belongs in a certain class based on the listed actions, explain briefly. Draw the precedence graph.

**a) T1:R(X), T2:R(X), T1:W(X), T2:W(X) (1 mark)**

**b) T1:W(X), T2:R(Y), T1:R(Y), T2:R(X) (1 mark)**

**c) T1:R(X), T1:R(Y), T1:W(X), T2:R(Y), T3:W(Y), T1:W(X), T2:R(Y) ( 2 marks)**

**d) T1:W(X), T2:R(X), T1:W(X), T2:Abort, T1:Commit (1 mark)**

**(marks: 5)**

(ii) Is the following Schedule conflict serializable? Draw Precedence graph.

**S: R3(y) R3(z) R1(x) W1(x) W3(y) W3(z) R2(z) R1(y) W1(y) R2(y) W2(y) R2(x) W2(x)**

**(marks: 5)**

**Total marks: 10**

2. Consider the following classes of schedules: serializable, conflict-serializable, view-serializable, recoverable, avoids-cascading-aborts, and strict. For each of the following schedules, state which of the preceding classes it belongs to. If you cannot decide whether a schedule belongs in a certain class based on the listed actions, explain briefly. Draw the precedence graph.

**a) T1:W(X), T2:R(X), T1:W(X), T2:Commit, T1:Commit (1 mark)**

**b) T1:W(X), T2:R(X), T1:W(X), T2:Commit, T1:Abort (1 mark)**

**c) T2: R(X), T3:W(X), T3:Commit, T1:W(Y), T1:Commit, T2:R(Y), T2:W(Z), T2:Commit (1 mark)**

**d) T1:R(X), T2:W(X), T1:W(X), T3:R(X), T1:Commit, T2:Commit, T3:Commit (1 mark)**

ii) Consider the following schedules :

**S1: r1(w), w1(x), r1(y) , w1(y), r1(x), w2(x), c2, c1**

**S2: r1(x), w1(x), r2(x), r1(y), w2(x), w1(y), c1, c2**

Which schedule is recoverable explain in steps and graphs if needed? **(6 marks)**

**Total marks: 10**



a. Consider the following Schedule S having 4 transactions and are executed Concurrently. The order of their operations are as follows:

**S : r1(x), r2(y), w2(y), w3(x), r3(z), w3(y), r1(z), w4(x), w3(z), r4(x), r4(z), w4(z) c1, c2,**

**c3, c4**

Find if S is what serializable? Conflict or view? Draw graph if needed. **(6 marks)**

b. Is the following recoverable or strict or cascadeless schedule?

**r2(x), r1(x), w1(x), r1(y) ,w1(y), c1, w2(x), c2** **(1 mark)**

**r1(x), w1(x), r2(x), r1(y), w2(x), c2, a1**  **(1 mark)**

**r1(x), w1(x), r2(x), r1(y), w2(x), w1(y), a1, a2**  **(1 mark)**

**r1(x), r2(x), w1(x), r1(y), w1(y), c1, w2(x), c2**  **(1 mark)**

**Total marks: 10**

2. Consider the following code and Find out the what type of 2PL locking is used:
3. **Lock\_S(basic)**

**Lock\_X(netsal)**

**Lock\_S(deduct)**

**Lock\_X(hra)**

**R(Basic)**

**R(deduct)**

**Hra-hra\_hra\*0.5**

**Netsal = basic -deduct +hra**

**Unlock(basic)**

**Unlock(deduct)**

**W(netsal)**

**W(hra)**

**Commit**

**Unlock (netsal)**

**Unlock(hra)** **(2 marks)**

1. **Lock\_s(A)**

**R(A)**

**Lock\_X(B)**

**B= A +B**

**Unlock(B)**

**Unlock(A) (2 marks)**

1. Consider the following two transactions:



Add lock and unlock instructions to transactions T31 and T32, so that they observe the two-phase locking protocol. Can the execution of these transactions result in a deadlock? **(6 marks)**

**Total marks: 10**



i)Consider a variant of the tree protocol called the *forest* protocol. The database is organized as a forest of rooted trees. Each transaction *Ti* must follow the following rules:

* The first lock in each tree may be on any data item.
* The second, and all subsequent, locks in a treemay be requested only if the parent of the requested node is currently locked.
* Data items may be unlocked at any time.
* A data item may not be relocked by *Ti* after it has been unlocked by *Ti* .

Show that the forest protocol does *not* ensure serializability. **(6 marks)**

ii)

Consider the following two transactions:

***T*34: read(*A*);**

**read(*B*);**

**if *A* = 0 then *B* := *B* + 1;**

**write(*B*).**

***T*35: read(*B*);**

**read(*A*);**

**if *B* = 0 then *A* := *A* + 1;**

**write(*A*).**

a. Add lock and unlock instructions to transactions *T*31 and *T*32, so that they observe the two-phase locking protocol. **(2 marks)**

b. Can the execution of these transactions result in a deadlock? **(2 marks)**

**Total marks:10**

2. Consider the transactions t1, t2 and t3 and a schedule S given below.

S : **read**1(A); **read**2 (B); **write**1(C); **read**3(B); **read**3(C); **write**2 (B); **write**3(A) Where the subscript denotes the transaction number. Assume that the time stamp of t1<t2<t3.

Using time-stamp ordering scheme for concurrency control find out if the schedule will

go through. If there is to be a rollback, which transaction(s) will be rolled back?

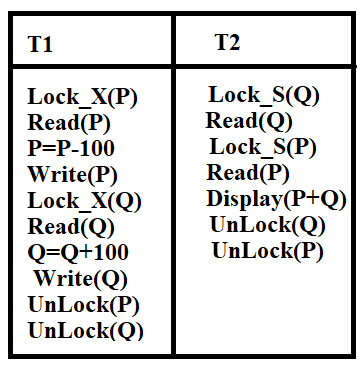
1. **marks)**

**b).** The following represents the sequence of events in a schedule involving transactions T1,

T2, T3, T4 and T5. A,B, C, D, E, F are items in the database. Draw a wait-for-graph for the data above and find whether the transactions are in a deadlock or not? **(3 marks)**



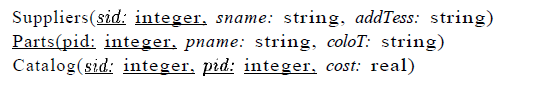
**c).** Consider the following code:



Is there is any deadlock? If so how do you modify it to avoid deadlock? **(4 marks)**

**Total marks:10**

2. Consider a database with the following schema:



The Catalog relation lists the prices charged for parts by Suppliers.

Consider three transactions *T1, T2,* and *T3.*

*T1* always has SQL isolation level SERIALIZABLE.

We first run *T1* concurrently with *T2* and then we run *T1* concurrently with *T2* but we change

the isolation level of *T2* as specified below. Give a database instance and SQL statements for

*T1* and T*2* such that result of running *T2* with the first SQL isolation level is different from

running T2 with the second SQL isolation level. Also specify the common schedule of *T1* and *T2.*

*Also* explain why the results are different.

1. SERIALIZABLE versus REPEATABLE READ. **(2.5 marks)**

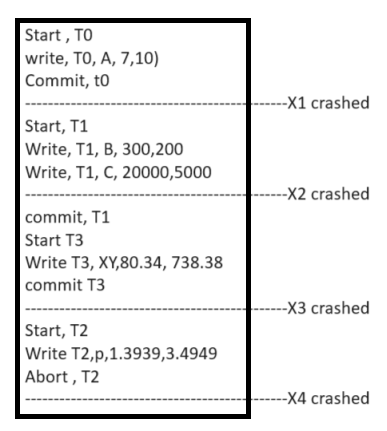
2. REPEATABLE READ versus READ COMMITTED. **(2.5 marks)**

**b.** Consider the following code and determine the

**a) redo and undo list contents (2 marks)**

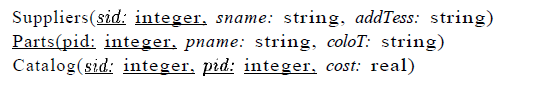
**b) each data item value. (2marks)**

**c) order of transactions for recovery (1marks)**



**Total marks:10**

2. Consider a database with the following schema:



The Catalog relation lists the prices charged for parts by Suppliers. For each of the following transactions, state the SQL isolation level that you would use and explain why you chose it.

1. A transaction that adds a new part to a supplier's catalog. **(2 marks)**

2. A transaction that increases the price that a supplier charges for a part. **(2 marks)**

3. A transaction that determines the total number of items for a given supplier. **(2 marks)**

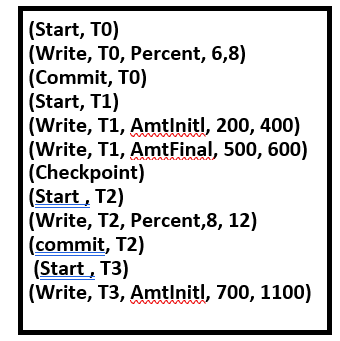
**Marks: 6**

b. Consider the following code and determine the

**i) redo and undo list contents (1.5 marks)**

**ii) each data item value. (1.5 marks)**

**iii) order of Transactions to recover. (1 marks)**



**Total marks:10**

2. We have the scenario as follows :

cust(cid, cname, amt) and key is cid, with the following two concurrent transactions.

**T1:**

**Begin Transaction**

**B1: update cust set amt= 13.5 \*amt where name = 'Bob'**

**B2: update cust set amt = 13.5 \* amt where name = 'Bob'**

**Commit**

**T2:**

**Begin Transaction**

**C1: update cust set amt = amt \* 0.5 where name = 'Bob'**

**C2: update cust set amt = amt \* 1.2 where name = 'Bob'**

**Commit**

Assume that the individual statements B1, B2, C1, and C2 always execute atomically. Bob's amt is 2000 before either transaction begins execution.   
(i) Suppose both transactions T1 and T2 execute to completion with isolation level Serializable. What are the possible values for Bob's final amt? **(2 marks)**  
(ii) Suppose both transactions T1 and T2 execute to completion with isolation level Read-Committed. What are the possible values for Bob's final amt? **(2 marks)**  
(iii) Suppose transaction T1 executes with isolation level Read-Committed, transaction T2 executes with isolation level Read-Uncommitted, and both transactions execute to completion. What are the possible values for Bob's final amt? **(2marks)**

1. Consider the following contents of a LOG file and determine the:

**i) redo and undo list contents (1.5 marks)**

**ii) each data item value. (1.5 marks)**

**iii) order of Transactions to recover. (1 marks)**

**(Start, T0)**

**(Write, T0, Interest, 6,8)**

**--------------------x1 crashed**

**(Commit, T0)**

**(Start, T1)**

**(Write, T1, Principle, 200, 400)**

**(Write, T1, Amount, 500, 600)**

**(Commit, T1)**

**(Start , T2)**

**(Write, T2, Interest, 6, 9)**

**--------------------x2 crashed**

**(commit, T2)**

**(Start , T3)**

**(Write, T3, Interest, 6, 11)**

**(Abort, T3)**

**Total marks:10**



**MID DataRecovery** are designing backend for Attendence-payroll systems at ACU University. They have to maintain employee records, different departments, Attendance , update payrolls and generate monthly pay slip. The users of DB are Employees who are the faculty , technical and non technical staffs, administrator, manager of each department and finally accountant of CA Uni.

You need to implement these:

|  |  |
| --- | --- |
| Employee | Can only see that data |
| Manager of a department | Changes the attendance if employee reports late for duty |
| Administrator | Maintains whole db |
| Accountant | Deals with salary of all employee and generates pay slip. |

What are all database objects are required like SP, Triggers, forms and others.

1. **Design the access matrix. (5 marks)**
2. **Implement the access matrix. (5 marks)**

**Total marks: 10 marks.**



Consider that all the relations were created by (and hence are owned by) user X, who wants to grant the following privileges to user accounts A, B, C, D, and E:

i) Account A can retrieve or modify any relation except DEPENDENT and can grant any of these privileges to other users.

ii) Account B can retrieve all the attributes of EMPLOYEE and DEPARTMENT except for Salary, Mgr\_ssn, and Mgr\_start\_date.

iii) Account C can retrieve or modify WORKS\_ON but can only retrieve the Fname, Minit, Lname, and Ssn attributes of EMPLOYEE and the Pname and Pnumber attributes of PROJECT.

iv) Account D can retrieve any attribute of EMPLOYEE or DEPENDENT and can modify DEPENDENT.

v) Account E can retrieve any attribute of EMPLOYEE but only for EMPLOYEE tuples that have Dno = 3.

vi) Write SQL statements to grant these privileges. Use views where appropriate.

1. **Design the access matrix. (5 marks)**
2. **Implement the access matrix.** (**5 marks)**

**Total marks: 10 marks.**



**DataShare** are designing backend for an Customer systems at JK ISP company which has huge client base and they cater to broadband, mobile phone connections .

They have to maintain customer records, different packages for each service , Bill reminders, technical issues and complaints and generate monthly bill. There are some people who would use the DB and they are manager, senior tech, supervisor and customer service person of JK.

You should implement these:

|  |  |
| --- | --- |
| Manager | Can make changes if needed |
| Senior Tech | Fixes the technical issues in any service and update the db |
| Supervisor | Can correct the bill if wrongly computed or if customers concern is legit |
| Customer service person | Deals with customer query and so this person can only view data. |

What all database objects are required like SP, Triggers, forms and others.

**i). Design the access matrix. (5 marks)**

**ii). Implement the access matrix.** (**5 marks)**

**Total marks: 10 marks.**

====================================XXXXXXXXXX=================================