Department: Computer Science and Engineering
Course Name: Database **Code: CS 354**
Full Marks-100 **Time: 3 hours**

Make reasonable assumptions as and whenever necessary. Answer the questions to any sequence but answers to all the parts of any question should appear together. Marks will be deducted if this is not followed properly.

Q 1. a) Explain the distinction between the terms serial schedule and serializable schedule. 3

b) List the ACID properties. Explain the usefulness of each. 2+ 4

c) Consider the following two transactions:

T13: read(A)
Read(B);
If A=0 then B=B+1;
Write(B);
T14: read(B);
Read(A);
If B=0 then A=A+1;
Write(A);

Let the consistency requirement be A=0 v B=0 with A=B=0 the initial values.

i) Show that every serial execution involving these two transactions preserves the consistency of the database.
ii) Show a concurrent execution of T13 and T14 that produces a nonserializable schedule.
iii) Is there a concurrent execution of T13and T14 that produces a serializable schedule.

3+3+3

d) During its execution, a transaction passes through several states, until it finally commits or aborts. List all possible sequences of states through which a transaction may pass. Explain why each state transition may occur.

5

Q2. a) Consider the following log records for the transactions T1, T2, T3 and T4. Describe the recovery process from a system crash if immediate update technique is followed. Also assume that the system maintains the checkpoints. Determine the list of transactions that need to be undone and that need to be redone. Also find the value of the data items X, Y, Z, U, V and W on disk after recovery.

(i) if the system crashes just before line 12 is written to the disk.
(ii) if the system crashes just before line 17 is written to the disk.

1. <START T1>
2. <T1, X, 30, 10>
3. <T1, Y, 50, 0>
4. <START T2>
5. <T1, X, 50, 30>
6. <T2, Z, 30, 10>
7. <COMMIT T1>

8. <START T3>
9. <T3, U, 70, 30>
10. <T2, V, 40, 25>
11. <T2, Z, 45, 30>
12. <COMMIT T2>
13. <CHECKPOINT>

14. <START T4>
15. <T4, W, 90, 10>
16. <COMMIT T3>
17. <T4, W, 100, 80>
18. <CHECKPOINT>
19. <COMMIT T4>

The log entries for database updates are in the format
<Transaction Id, Variable, New value, Old value> 3+3

b) What is Deadlock? What are the different ways of handling deadlock? 2+3

c) Assume a read-lock is requested before each read, and a write lock before each write. All unlocks occur after the last operation of a transaction. Explain what operations are denied during each schedule, draw the wait-for graph, and pick a transaction to abort if a deadlock does occur.

i) r1(A); r2(B); w1(C); r3(D); r4(E); w3(B); w2(C); w4(A); w1(D);

3

d) Define starvation.

2

Q3. a) Four transactions are presently running. (The abbreviation R(A) means Read(A), and so on)

T1: R(A),R(B),R(J),W(A),W(B),R(D),R(E),R(G)
T2: R(C),R(G),W(C),R(H),W(H),R(D)
T3: W(J),R(C),W(C)
T4: R(G),W(D),R(J)

The following is a schedule for these transactions.

T1:R(A), T4:R(G), T3:W(J), T1:R(B), T1:R(J), T1:W(A), T3:R(C), T3:W(C), T2:R(C), T2:R(G),
T1:W(B), T1:R(D), T2:W(C), T1:R(E), T1:R(G), T2:R(H), T2:W(H), T4:W(D), T4:R(J), T2:R(D)

i. List all conflicts in these transactions.
ii. Is the Schedule Conflict Serializable?. If it is, to which serial schedule(s) is it equivalent? Use your Serializability graph to justify your answer. 5+5

b) Consider a RAID Level 5 organization comprising six disks, with the parity for sets of five blocks on five disks stored on the sixth disk. How many blocks are accessed in order to perform the following?

i. A write of one block of data
ii. A write of seven continuous blocks of data 2.5 + 2.5 = 5

Q4. a) B+-trees are often used as index structures for database files because they maintain their efficiency despite repeated insertion and deletion of data.

(i) Show the structure of a B+-tree for a file containing records with the following search key values, assuming that the tree is initially empty, that three pointers fit in one node, and that records are added in the order given: Sundin, Fleury, Bure, Lindros, Federov
(ii) Now show the structure of the B+-tree from part (i) after the insertion of a record with the search key value 'Yashin'.

5+3

b) Suppose that extendable hashing is being used on a database file that contains records with the following search key values: 2, 3, 5, 7, 11, 17, 19, 23, 29, 31
(i) Construct the extendable hash structure for this file if the hash function is h(x) = x mod 7 and each bucket can hold three records.
(ii) Show how the structure from part (i) changes after inserting a record with the search key value of 16 and then deleting the record with the search key value of 11.

5+5

c) Differentiate between sparse and dense indexing. 3

d) Consider a disk pack with a seek time of 4 milliseconds and rotational speed of 10000 rotations per minute (RPM). It has 600 sectors per track and each sector can store 512 bytes of data. Consider a file stored in the disk. The file contains 2000 sectors. Assume that every sector access necessitates a seek, and the average rotational latency for accessing each sector is half of the time for one complete rotation. Calculate the total time (in milliseconds) needed to read the entire file. 5

Q5) a) What are the basic steps of query processing? What are the different components of a query cost? Explain the components.

3+3

b) Write the nested-loop join algorithm. Analyze total block-transfers and seeks required for nested-loop join approach. 3+4

c) How many of the following statements are true? 4

i) Deadlock is not possible with timestamp protocols.
ii) A transaction that arrives later to the system always has a smaller timestamp.
iii) The precedence graph for the timestamp algorithm has edges from smaller timestamp transactions to larger ones.
iv)A write is only performed if transaction has a timestamp >= the read timestamp for the data item.

d) Indicate what happens during each of these schedules where concurrency control is performed using timestamps: 3

i) st1; st3; st2; r1(A); r2(B); w1(C); r3(B); r3(C); w2(B); w3(A);