

**ASSIGNMENT SHEET 1**

- Q1.** Discuss the advantages of database system over file processing system.
- Q2.** Explain the three schema architecture of DBMS. Give suitable example to represent the three levels.
- Q3.** What do you understand by the term data abstraction? State the significance of mapping between the various levels of three level architecture of DBMS.
- Q4.** What are the different types of database users who interact with database system? List the responsibilities of the DBA.
- Q5.** Draw the overall structure of DBMS and state the functionality of the different components of the structure.
- Q6.** Discuss the candidate key, primary key, super key, alternate key, and secondary key.
- Q7.** Explain specialization, generalization and aggregation. Give suitable example in support of your answer.
- Q8.** Draw an E-R diagram of the book club. The book club has members. The book club sales books to its members. The member places orders for books, which the book club fulfills. Each order contains one or more than one books. The books are written by author(s). The publisher publishes the books and the author can write more than one book and a book can have more than one author. A book is published by publisher, but a publisher publishes many books. A member can place more than one order. He also can choose not to place an order. The book club sales many books.
- Q9.** What is the purpose of the E-R diagram? Construct an E-R diagram for a university system which should include information about students, departments, professors, courses, which students are enrolled in which course, which professors are teaching which courses, student grades, which course a department offers. Make suitable assumptions wherever required.
- Q10.** Construct an E-R diagram for a hospital with a set of patients and a set of medical doctors. Associate with each patient a log of the various tests and examinations conducted.

**ASSIGNMENT SHEET 2**

**Q1.** Consider the following relation schemes:

Project (Project#, Project\_name, chief\_architect)

Employee (Emp#, Empname)

Assigned\_To (Project#, Emp#)

Give expression in Tuple calculus and Domain calculus for each of the queries below:

- (i) Get the employee numbers of employees who work on all projects.
- (ii) Get the employee numbers of employees who do not work on the COMP123 project.

**Q2.** Define the five basic operators of relational algebra with an example each.

**Q3.** Explain entity integrity and referential integrity rules in relational model. Show how these are realized in SQL.

**Q4.** Consider the following relations:

S (S#, SNAME, STATUS, CITY)

SP (S#, P#, QTY)

P (P#, PNAME, COLOR, WEIGHT, CITY)

Give an expression in SQL for each of queries below:

- (i) Get supplier names for supplier who supply at least one red part
- (ii) Get supplier names for supplier who do not supply part P2.

**Q5.** Define a view and a trigger. Construct a view for the above relations which has the information about suppliers and the parts they supply. The view contains the S#,SNAME, P# , PNAME renamed as SNO, NAME, PNO, PNAME.

**Q6.** Differentiate between the following: (i) Theta Join. (ii) Equi Join. (iii) Natural Join (iv) Outer Join.

**Q7.** Explain the integrity constraints: Not Null, Unique, Primary Key with an example each. Is the combination 'Not Null, Primary Key' a valid combination. Justify.

**Q8.** Explain the followings :

- (i) Nested Queries.
- (ii) Cursors in SQL.

**Q9.** Consider the following relational schemas:

EMPLOYEE (EMPLOYEE\_NAME, STREET, CITY)

WORKS (EMPLOYEE\_NAME, COMPANYNAME, SALARY)

COMPANY (COMPANY\_NAME, CITY)

Specify the table definitions in SQL.

**Q.10** Give an expression in SQL for each of queries below:

- (i) Find the names of all employees who work for first Bank Corporation .(ii) Find the names and company names of all employees sorted in ascending order ofcompany name and descending order of employee names of that company.(iii) Change the city of First Bank Corporation to 'New Delhi'

ASSIGNMENT SHEET 3

**Q1.** Suppose you are given a relation  $R = (A,B,C,D,E)$  with the following functional dependencies:  $\{CE \rightarrow D, D \rightarrow B, C \rightarrow A\}$ .

- Find all candidate keys.
- Identify the best normal form that  $R$  satisfies (1NF, 2NF, 3NF, or BCNF).
- If the relation is not in BCNF, decompose it until it becomes BCNF. At each step, identify a new relation, decompose and re-compute the keys and the normal forms they satisfy.

**Q2.** You are given the following set of functional dependencies for a relation  $R(A,B,C,D,E,F)$ ,  $F = \{AB \rightarrow C, DC \rightarrow AE, E \rightarrow F\}$ .

- What are the keys of this relation?
- Is this relation in BCNF? If not, explain why by showing one violation.
- Is the decomposition  $(A,B,C,D)$   $(B,C,D,E,F)$  a dependency preserving decomposition? If not, explain briefly.

**Q3.** You are given the below set of functional dependencies for a relation  $R(A,B,C,D,E,F,G)$ ,  $F = \{AD \rightarrow BF, CD \rightarrow EGC, BD \rightarrow F, E \rightarrow D, F \rightarrow C, D \rightarrow F\}$ .

- Find the minimal cover for the above set of functional dependencies using the algorithm described in class. Give sufficient detail to show your reasoning, but be succinct. You do not have to list all the cases you test/consider for the algorithm. Show all steps where you make changes to the above set in detail.
- Using the functional dependencies that you computed in step a, find the keys for this relation. Is it in BCNF? Explain your reasoning.
- Suppose we decompose the above relation into the following two relations:  $R_1(A,B,C,D,E)$   $R_2(A,D,F,G)$  Use the functional dependencies in the minimal cover. For each relation, write down the functional dependencies that fall within that relation (you can decompose a dependency of the form  $AD \rightarrow BF$  into two i.e.  $AD \rightarrow B$  and  $AD \rightarrow F$  when computing this).

**Q4.** Given relation  $R(W, X, Y, Z)$  and set of functional dependencies  $F = \{X \rightarrow W, WZ \rightarrow XY, Y \rightarrow WXZ\}$ . Compute the minimal cover for  $F$ .

**Q5.** Given  $R(A,B,C,D, E)$  with the set of FDs,  $F(AB \rightarrow CD, A \rightarrow E, C \rightarrow D)$ . Is the decomposition of  $R$  into  $R_1(A,B,C)$ ,  $R_2(B,C,D)$  and  $R_3(C,D,E)$  lossless? Prove.

**Q6.** What do you understand by transitive dependencies? Explain with an example any two problems that can arise in the database if transitive dependencies are present in the database.

**Q7.** List the Armstrong's axioms for functional dependencies. What do you understand by soundness and completeness of these axioms?

**Q8.** Consider the decomposition of relation scheme, shipping = (Ship, Capacity, Date, Cargo, Value) with the set of functional dependencies,  $F = \{Ship \rightarrow Capacity, Ship, Date \rightarrow Cargo, Cargo, Capacity \rightarrow Value\}$  into  $R_1 = \{Ship, Capacity\}$  with  $F_1 = \{Ship \rightarrow Capacity\}$  and  $R_2 = \{Ship, Date, Cargo, Value\}$  with  $F_2 = \{Ship, Date \rightarrow Cargo, Value\}$ . Is this decomposition in BCNF? Is this decomposition lossless and dependency preserving? Justify your answers.

**Q9.** Define functional and multi-valued dependencies.

**Q10.** What are the three data anomalies that are likely to occur as a result of data redundancy? Can data redundancy be completely eliminated in database approach? Why or why not?

ASSIGNMENT SHEET 4

**Q1.** List the ACID properties. Explain the usefulness of each.

**Q2.** 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.

**Q3.** Justify the following statement: Concurrent execution of transactions is more important when data must be fetched from (slow) disk or when transactions are long, and is less important when data is in memory and transactions are very short.

**Q4.** Consider the following two transactions:

T1: read(A); read(B); if A = 0 then B := B + 1; write(B).

T2: read(B); read(A); if B = 0 then A := A + 1; write(A).

Let the consistency requirement be  $A = 0 \vee B = 0$ , with  $A = B = 0$  the initial values. a. Show that every serial execution involving these two transactions preserves the consistency of the database. b. Show a concurrent execution of T1 and T2 that produces a nonserializable schedule. c. Is there a concurrent execution of T1 and T2 that produces a serializable schedule?

**Q5.** What is a cascadeless schedule? Why is cascadelessness of schedules desirable? Are there any circumstances under which it would be desirable to allow noncascadeless schedules? Explain your answer.

**Q6.** Compare the shadow-paging recovery scheme with the log-based recovery schemes in terms of ease of implementation and overhead cost.

**Q7.** When the system recovers from a crash (see Section 17.6.4), it constructs an undo-list and a redo-list. Explain why log records for transactions on the undolist must be processed in reverse order, while those log records for transactions on the redo-list are processed in a forward direction.

**Q8.** Assume that immediate modification is used in a system. Show, by an example, how an inconsistent database state could result if log records for a transaction are not output to stable storage prior to data updated by the transaction being written to disk

**Q9.** Compare the deferred- and immediate-modification versions of the log-based recovery scheme in terms of ease of implementation and overhead cost.

**Q10.** Explain the purpose of the checkpoint mechanism. How often should checkpoints be performed? How does the frequency of checkpoints affect

- System performance when no failure occurs
- The time it takes to recover from a system crash
- The time it takes to recover from a disk crash

### ASSIGNMENT SHEET 5

**Q1.** Consider the following two transactions: T31: read(A); read(B); if A = 0 then B := B + 1; write(B). T32: read(B); read(A); if B = 0 then A := A + 1; write(A).

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?

**Q2.** Consider the validation-based concurrency-control scheme of Section 16.3. Show that by choosing Validation( $T_i$ ), rather than Start( $T_i$ ), as the timestamp of transaction  $T_i$ , we can expect better response time provided that conflict rates among transactions are indeed low.

**Q3.** For each of the following protocols, describe aspects of practical applications that would lead you to suggest using the protocol, and aspects that would suggest not using the protocol: • Two-phase locking • Two-phase locking with multiple-granularity locking • The tree protocol • Timestamp ordering • Validation • Multiversion timestamp ordering • Multiversion two-phase locking.

**Q4.** Explain the phantom phenomenon. Why may this phenomenon lead to an incorrect concurrent execution despite the use of the two-phase locking protocol?

**Q5.** Show that there are schedules that are possible under the two-phase locking protocol, but are not possible under the timestamp protocol, and vice versa.

**Q6.** Discuss the relative advantages of centralized and distributed databases.

**Q7.** Give an example where the read one, write all available approach leads to an erroneous state.

**Q8.** Stable storage cannot be implemented. a. Explain why it cannot be. b. Explain how database systems deal with this problem.

**Q9.** Explain the difference between the three storage types—volatile, nonvolatile, and stable—in terms of I/O cost.

**Q10.** Use of multiple-granularity locking may require more or fewer locks than an equivalent system with a single lock granularity. Provide examples of both situations, and compare the relative amount of concurrency allowed.

