Sardar Patel Institute of Technology
Bhavan's Campus, Munshi Nagar, Andheri (West), Mumbai-400058
(An Autonomous Institute Affiliated to University of Mumbai)

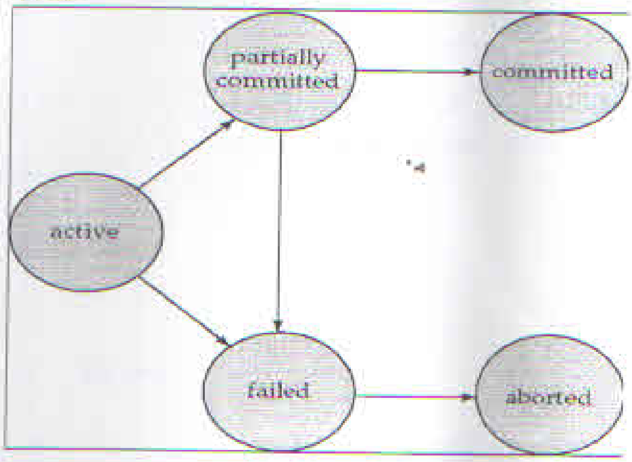
End Semester Examination

April 2018

Max. Marks: 100 **Duration: 180 Minutes**
Class: SE **Semester: 4**
Course Code: CE42, IT43
Branch: COMP, IT
Name of the Course: Database Management System

Instructions:
(1) **All Questions are Compulsory**
(2) **Draw neat diagrams**
(3) **Assume suitable data if necessary**

Question No.		Max. Marks	CO
Q1	Answer any 4 out of 6		
(a)	Discuss the advantages of Database over File Systems. Soln: • Data redundancy and inconsistency • Data isolation • Integrity problems • Atomicity problems • Concurrent access anamolies	05	1
(b)	Consider the following table. A | B | C 10 | b1 | c1 10 | b2 | c2 11 | b4 | c1 12 | b3 | c4 13 | b1 | c1 14 | b3 | c4 Which of the following dependencies may hold in the above relation? If the dependency cannot hold, explain which tuple causes violation. i. B→C ii. A →B iii. B→A iv. C→A v. C→B Soln: The following dependencies will hold in the above relation B→C C→B The dependencies, A→B, B→A, C→A ,will not hold in the above relation as they will not uniquely identify a tuple	1+1+3	4

(c)	Explain generalization and specialization with appropriate examples. Soln: • The process of designating subgroupings within an entity set. • Depicted by a *triangle* component labeled ISA (E.g. *customer* "is a" *person*). • **Attribute inheritance** – a lower-level entity set inherits all the attributes and relationship participation of the higher-level entity set to which it is linked. • The ISA relationship also referred to as **superclass - subclass** relationship • example	3+2	
(d)	Discuss why assertions must be used with utmost care. Soln: • An assertion is a predicate expressing condition that we wish the database always satisfies. • An assertion in SQL takes the form: Create assertion <assertion-name> check <predicate> • When an assertion is made, the system tests its validity and tests it again on every update that may violate the assertion. • This testing may introduce a significant amount of overhead. Hence, assertions must be used with care. • Example of assertion	1 1 2 1	3
(e)	Illustrate with the help of a diagram the various transaction states. Soln:  **Active** – the initial state; the transaction stays in this state while it is executing	2+3	5

	Partially committed – after the final statement has been executed. **Failed** -- after the discovery that normal execution can no longer proceed. **Aborted** – after the transaction has been rolled back and the database restored to its state prior to the start of the transaction. Two options after it has been aborted: • **restart the transaction**: can be done only if hardware or software error. • **kill the transaction**: It is usually done because of some internal logical error that can be corrected only by rewriting the application program. **Committed** – after successful completion.		
(f)	Explain the role of DBA Soln: 1. Schema definition 2. storage structure and access method definition 3. schema and physical organization modification 4. granting of authorization for data access 5. routine maintenance	05	1
Q2 (a)	Draw an ER diagram for Library Management System and map the ER diagram to relations. Soln: The ER diagram must include all the concepts like entity, weak entity, relationships, mapping cardinalities, types of attributes, participation constraints. Mapping to tables	6+4	2
Q2 (b)	Consider the relational database given below, where the primary keys are underlined. employee (person-name, street, city) works (person-name, company-name, salary) company (company-name, city) manages (person-name, manager-name) Give an expression in the relational algebra to express each of the following queries: i. Find the names of all employees who work for First Bank Corporation. Soln: Πperson-name (σcompany-name = "First Bank Corporation" (works)) ii. Find the names and cities of residence of all employees who work for First Bank Corporation.	2*5	2

	Soln: Πperson-name, city (employee (σcompany-name = "First Bank Corporation" (works)) iii. Find the names, street address, and cities of residence of all employees who work for First Bank Corporation and earn more than $10,000 per annum. Soln: c. Πperson-name, street, city (σ(company-name = "First Bank Corporation" ∧ salary > 10000) works employe iv. Find the names of all employees in this database who live in the same city as the company for which they work. Soln: Πperson-name (employee works company) v. Assume the companies may be located in several cities. Find all companies located in every city in which Small Bank Corporation is located. Soln: Πcompany-name (company ÷ (Πcity (σcompany-name = "Small Bank Corporation" (company)))			
Q3 (a)	What is a view in SQL? How is it defined? Discuss the problems that may arise while updating a view. Soln: In SQL, a view is a virtual table based on the result-set of an SQL statement. A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database. Syntax: CREATE VIEW view_name AS SELECT column1, column2, ... FROM table_name WHERE condition; Example Problems in modification OR Explain referential integrity with example. Soln: definition syntax on delete/update: no action, set null, set default, cascade	2+2+6 2+2+4+2	3	

	example		
3 (b)	Consider the following schema and solve the queries Course (course_no, title) Offering (course_no, off_to, off_date, location) Teacher (course_no, off_no, emp_no) Enrolment (course_no, off_no, stud_no, grade) Employee (emp_no, emp_name, job, sal) Student (stud_no, stud_name, phone_no) 1. List all the teachers who conduct the course titled "Database Systems" Soln: select empno from teacher where course_no =(Select course_no from course where title like 'Database Systems') 2. List all the courses offered in 'Thane' on 15/8/15. Soln: Select course_no from offering where location like 'thane'and off-date like '1t-aug-2015' 3. Find the courses enrolled by "Monali". Soln: Select course-no from enrolment where stud_no = (Select stud_no from student where stud_name like 'monali') 4. List all the employees jobwise having average salary greater than 5000. Soln: Select job, count(*) From employee Groupby job Having avg(sal)>5000; 5. Add the date_of_birth attribute in the student table Soln: Alter table student add column date_of_birth varchar2(10);	5*2	3
Q4 (a)	What are triggers? Explain with example. Soln: • Define trigger • Explain ECA model • Explain row level and statement level trigger • Syntax • example	2+2+2+2+2	4

Q4 (b)

Consider the relation R(A,B,C,D,E,F,G,H,I,J) with following functional dependencies:

AB→C **BD →EF** **AD →GH** **A→I** **H→J**

Determine the candidate key for R and normalize the relation to 3rd normal form. Justify every step.

6+2+2

Soln:

{A, B}+= {A, B, C, I}
{B, D+= {B, D, E, F}
{A, D}+= {A, D, G, H, I, J}
{A}+= {A, I}
{H}+= {H, J}
{A, B, D}+= {A, B, C, D, E, F, G, H, I, J}
The key of the relation R is ABD

2NF

r1

A	B	C

r2

B	D	E	F

r3

A	D	G	H	J

r4

A	I

3NF

r3 violates 3NF therefore decompose as follows

r31

A	D	G	H

r32

H	J

The final tables after 2NF and 3NF are r1, r2, r31, r32, r4

OR

Consider the relation R(M,Y,P,MP,C) with following functional dependencies:

M,Y→P **M →MP** **MP →C**

Determine the candidate key for R and normalize the relation to 3rd normal form. Justify every step.

Soln:

{M,Y}+= {M,Y,P,MP}
{M}+= {M,MP,C}
{MP}+= {MP,C}

2NF:

TABLE 1: M,Y,P,MP
TABLE 2: M,MP,C

3+4+3

3NF:

Sardar Patel Institute of Technology
Bhavan's Campus, Munshi Nagar, Andheri (West), Mumbai-40005
(An Autonomous Institute Affiliated to University of Mumbai)

	TABLE 1.1 : M,Y,P TABLE 1.2 : M,MP TABLE 2.1: MP, C		
Q5	Answer any 4 out of 6		
	a. Deadlock with wait-for graph. Soln: A system is in a **deadlock** if it there exists a set of transactions such that every transaction in the set is waiting for another transaction in the set. Deadlocks can be described as a *wait-for graph*, which consists of a pair $G = (V,E)$, ★ V is a set of vertices (all the transactions in the system) ★ E is a set of edges; each element is an ordered pair $T_i \rightarrow T_j$. If $T_i \rightarrow T_j$ is in E, then there is a directed edge from T_i to T_j, implying that T_i is waiting for T_j to release a data item. When T_i requests a data item currently being held by T_j, then the edge T_i T_j is inserted in the wait-for graph. This edge is removed only when T_j is no longer holding a data item needed by T_i. The system is in a deadlock state if and only if the wait-for graph has a cycle. Must invoke a deadlock-detection algorithm periodically to look for cycles. Diagram for with and without wait for graph	2+3	5
	b. What is the meaning of serializability? How do you check whether a schedule is conflict serializable or not? Soln: • Consists of a sequence of instructions from various transactions where the instructions belonging to one single transaction appear together in that schedule. • For a set of 'n' transactions there can be 'n!' different valid serial schedules. • A schedule is conflict serializable if and only if its precedence graph is acyclic. • Cycle-detection algorithms exist which take order n2 time, where n is the number of vertices	2+3	

		in the graph.
		• If precedence graph is acyclic, the serializability order can be obtained by a topological sorting of the graph.
		• Example
		
		c. Shadow Paging technique
		Soln:
		• **Shadow paging** is an alternative to log-based recovery; this scheme is useful if transactions execute serially
		• Idea: maintain *two* page tables during the lifetime of a transaction –the **current page table**, and the **shadow page table**
		• Store the shadow page table in nonvolatile storage, such that state of the database prior to transaction execution may be recovered.
		  ○ Shadow page table is never modified during execution
		• To start with, both the page tables are identical. Only current page table is used for data item accesses during execution of the transaction.
		• Whenever any page is about to be written for the first time
		  ○ A copy of this page is made onto an unused page.
		  ○ The current page table is then made to point to the copy
		  ○ The update is performed on the copy
		• To commit a transaction :
		• Flush all modified pages in main memory to disk
		• Output current page table to disk
		• Make the current page table the new shadow page table, as follows:
		  ○ keep a pointer to the shadow page table at a fixed (known) location on disk.
		  ○ to make the current page table the new shadow page table, simply update the pointer to point to current page table on disk
		• Once pointer to shadow page table has been written, transaction is committed.
		• No recovery is needed after a crash — new transactions can start right away, using the shadow page table.
		• Pages not pointed to from current/shadow page table should be freed (garbage collected).
		
		d. Timestamp ordering protocol
		Soln:
		• Suppose a transaction Ti issues a read(Q)
			5
			
			5	

	1. If TS(Ti) ≤W-timestamp(Q), then Ti needs to read a value of Q that was already overwritten. Hence, the read operation is rejected, and Ti is rolled back. 2. If TS(Ti) ≥ W-timestamp(Q), then the read operation is executed, and R-timestamp(Q) is set to max(R-timestamp(Q), TS(Ti)). • Suppose that transaction Ti issues write(Q). 1. If TS(Ti) < R-timestamp(Q), then the value of Q that Ti is producing was needed previously, and the system assumed that that value would never be produced. Hence, the write operation is rejected, and Ti is rolled back. 2. If TS(Ti) < W-timestamp(Q), then Ti is attempting to write an obsolete value of Q. Hence, this write operation is rejected, and Ti is rolled back. 3. Otherwise, the write operation is executed, and W-timestamp(Q) is set to TS(Ti). e. Two phase locking protocol Soln: • Shared mode lock • Exclusive mode lock • Growing phase • Shrinking phase • example f. Validation based protocol Soln: 1. definition 2. phases of protocol 3. three timestamps associated 4. validation test 5. example	2+2+2+2+2 2+2+2+2+2	