USN | | | | | | | | | |

Internal Assessment Test 1 – June 2024

Sub:	Database Management Systems					Sub Code:	BCS403	Branch:	AIML/CSE-AIML
Date:	04/06/24	Duration:	90 minutes	Max Marks:	50	Sem/Sec:	IV		OBE

		SCHEME AND SOLUTIONS	MARKS	CO	RBT
1	a	Discuss the main characteristics of the database approach. How does it differ from traditional file system? **Answer:-** **Explanation of 4 points- 4X2=8M** • Self-describing nature of a database system • Insulation between programs and data, and data abstraction • Support of multiple views of the data • Sharing of data and multiuser transaction processing **Difference with traditional file system -2M**	[10]	1	L1
2	a	Construct an ER diagram for Hospital data base considering at least four entities and assume your own attributes and relationships **Answer: -** **Selection of 4 entities related to Hospital-2M** **Diagram with proper notation of entities, attributes, relationship, cardinality ratio and participation constraints-4M**	[6]	1	L3
	b	Distinguish between logical data independence and physical data independence? Which one is harder to achieve? Why? **Answer:** **Difference between Logical and Physical data independence-2M** **Which one is harder to achieve? Why?-2M** Achieving logical data independence is harder compared to physical data independence. Changes at the physical level do not need to be reflected at the application level. Changes at the logical level need to be reflected at the application level.	[4]	1	L2
3	a	Explain the concept of a data model? Explain different categories of data models. **Answer:-** **Data model Definition -2M** A data model—a collection of concepts that can be used to describe the structure of a database—provides the necessary means to achieve this abstraction **Different categories** **1.High-level or conceptual data models ----1M** **2.Low-level or physical data models---- 1M** **3, Representational (or implementation) data models---2M**	[6]	1	L2

	b	Explain the importance of Cardinality ratio and participation Constraints in a relational model. **Answer-** **Cardinality Ratio- Definition +Example –2M** The cardinality ratio for a binary relationship specifies the maximum number of relationship instances that an entity can participate in. **Participation Constraint- Definition+Example—2M** The participation constraint specifies whether the existence of an entity depends on its being related to another entity via the relationship type This constraint specifies the minimum number of relationship instances that each entity can participate in and is sometimes called the minimum cardinality constraint There are two types of participation constraints—total and partial	[4]	1	L2
4	a	Briefly describe the steps involved in ER- to relational mapping algorithm. **Answer:** **Mention the steps-2M** **Each step +Explanation -8M** Step 1: For each regular (strong) entity type E in the ER schema, create a relation R that includes all the simple attributes of E. Step 2: For each weak entity type W in the ER schema with owner entity type E, create a relation R, and includeall simple attributes (or simple components of composite attributes) of W as attributes. In addition, include as foreign key attributes of R the primary key attribute(s) of the relation(s) that correspond to the owner entity type(s). Step 3: For each binary 1:1 relationship type R in the ER schema, identify the relations S and T that correspond to the entity types participating in R. Choose one of the relations, say S, and include the primary key of T as a foreign key in S. Include all the simple attributes of R as attributes of S. Step 4: For each regular binary 1:N relationship type R identify the relation (N) relation S. the primary key of T as a foreign key of S. Simple attributes of R map to attributes of S. Step 5: For each binary M:N relationship type R, create a relation S. Include the primary keys of participant relations as foreign keys in S. Their combination will be the primary key for S. Simple attributes of R become attributes of S. Step 6: For each multi-valued attribute A, create a new relation R. This relation will include an attribute corresponding to A, plus the primary key K of the parent relation (entity type or relationship type) as a foreign key in R. The primary key of R is the combination of A and K. Step 7: For each n-ary relationship type R, where n>2, create a new relation S to represent R. Include the primary keys of the relations participating in R as foreign keys in S. Simple attributes of R map to attributes of S.	[10]	2	L2

		The primary key of S is a combination of all the foreign keys that reference the participants that have cardinality constraint > 1. For a recursive relationship, we will need a new relation.			
5	a	Explain the different schema-based constraints. **Answer:-** **List the constraints-1M** **Explanation of each type -1X5=5M** The schema-based constraints include domain constraints, key constraints, constraints on NULLs, entity integrity constraints, and referential integrity constraints.	[6]	2	L2
	b	Consider the relations. EMPLOYEE(emp_id,name) ASSIGNED_TO(projectno,emp_id) PROJECT(projectno,project_name) Express the following queries in Relational Algebra. i)Get details of employees working on both P354 and P345 project numbers. ii)Find the employee number of employee who work on project P678 **Answers: Each query 2M each** i)Π emp_id,name(∂project_no.=P354 (EMPLOYEE*ASSIGNED_TO)) ∩ Π emp_id,name (∂project_no.=P345 (EMPLOYEE*ASSIGNED_TO)) ii) Π emp_id(∂project_no.=P678 (EMPLOYEE*ASSIGNED_TO))	[4]	2	L3
6	a	List the various operations of relational algebra and explain the purpose of each with examples. **Answer:-** **Each operator=1X10=10M with examples.**	[10]	2	L2

Operation	Purpose	Notation
SELECT	Selects all tuples that satisfy the selection condition from a relation *R*.	$\sigma_{\text{<selection condition>}}(R)$
PROJECT	Produces a new relation with only some of the attributes of *R*, and removes duplicate tuples.	$\pi_{\text{<attribute list>}}(R)$
THETA JOIN	Produces all combinations of tuples from R_1 and R_2 that satisfy the join condition.	$R_1 \bowtie_{\text{<join condition>}} R_2$
EQUIJOIN	Produces all the combinations of tuples from R_1 and R_2 that satisfy a join condition with only equality comparisons.	$R_1 \bowtie_{\text{<join condition>}} R_2$, OR $R_1 \bowtie_{\text{(<join attributes 1>), (<join attributes 2>)}} R_2$
NATURAL JOIN	Same as EQUIJOIN except that the join attributes of R_2 are not included in the resulting relation; if the join attributes have the same names, they do not have to be specified at all.	$R_1 *_{\text{<join condition>}} R_2$, OR $R_1 *_{\text{(<join attributes 1>), (<join attributes 2>)}} R_2$ OR $R_1 * R_2$
UNION	Produces a relation that includes all the tuples in R_1 or R_2 or both R_1 and R_2; R_1 and R_2 must be union compatible.	$R_1 \cup R_2$
INTERSECTION	Produces a relation that includes all the tuples in both R_1 and R_2; R_1 and R_2 must be union compatible.	$R_1 \cap R_2$
SET DIFFERENCE	Produces a relation that includes all the tuples in R_1 that are not in R_2; R_1 and R_2 must be union compatible.	$R_1 - R_2$
CARTESIAN PRODUCT	Produces a relation that has the attributes of R_1 and R_2 and includes as tuples all possible combinations of tuples from R_1 and R_2.	$R_1 \times R_2$
DIVISION	Produces a relation $R(X)$ that includes all tuples $t[X]$	$R_1(Z) \div R_2(Y)$

CI **CCI** **HoD**

---All the Best---