Reg No.:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Pages: 5

Name:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

B.Tech Degree S4 (R,S) / S4 (PT) (R,S) Examination June 2023 (2019 Scheme)

### Part A Answers

1. **Six major advantages of using a DBMS:**
   * **Data redundancy and inconsistency reduction:** Multiple instances of the same data are minimized.
   * **Data access improvement:** Easier retrieval and manipulation of data for users.
   * **Data integrity and security enforcement:** DBMS enforces rules to maintain data accuracy and security.
   * **Data administration improvements:** Database administrators can manage data more effectively with tools provided by a DBMS.
   * **Concurrent access and crash recovery:** DBMS ensures data consistency in multi-access environments and recovers data from system crashes.
   * **Reduced application development time:** Offers tools and libraries that simplify the application development process.
2. **Concept of a weak entity in data modeling:**
   * **Weak Entity:** Relies on a foreign key from another (owner) entity type for its identity.
   * **Owner Entity Type:** The entity type on which a weak entity depends for its identity.
   * **Identifying Relationship Type:** The relationship through which a weak entity is associated with its owner, signifying dependence.
3. **Theta join explanation and example:**
   * **Theta Join:** A relational database operation that links tables based on a relationship other than the standard equi-join. It uses a conditional expression (theta).
   * **Example:**
     + Given relations R(A, B, C) and S(D, E) and theta as B < D.
     + Resulting tuples are those combinations of R and S where the value of B in R is less than the value of D in S.
4. **Definitions:**
   * **Primary Key:** A column, or set of columns, that uniquely identifies each row in a table.
   * **Candidate Key:** Any column, or set of columns, that can qualify as a unique key in database terms.
   * **Super Key:** A set of one or more columns that can uniquely identify a record in a table.
5. **Difference between WHERE and HAVING clauses:**
   * **WHERE Clause:** Used to filter records before any groupings are made.
   * **HAVING Clause:** Used to filter values after they have been groups.
   * **Example:** If you want to find departments that have more than five employees:
     + Use WHERE to filter records before grouping them by department.
     + Use HAVING to filter these groups after the count of employees per department is calculated.
6. **Difference between Hash indexes and B+-tree indexes:**
   * **Hash Indexes:** Best for equality searches, where the hash function computes the location of a data record given its key.
   * **B+-Tree Indexes:** Suitable for range and point queries, as they maintain data in sorted order.
7. **Functional dependency and triviality:**
   * **Functional Dependency:** A relationship that exists when one attribute uniquely determines another attribute.
   * **Trivial Functional Dependency:** Occurs when all attributes on the right-hand side of the dependency are a subset of the left-hand side, meaning the dependency is inherently true.
8. **Armstrong's Axioms:**
   * **Reflexivity:** If Y is a subset of X, then X → Y.
   * **Augmentation:** If X → Y, then XZ → YZ for any Z.
   * **Transitivity:** If X → Y and Y → Z, then X → Z.
9. **ACID properties of transactions:**
   * **Atomicity:** Ensures transactions are fully completed or not at all.
   * **Consistency:** Ensures the database remains in a consistent state after any transaction.
   * **Isolation:** Ensures transactions are processed independently of one another.
   * **Durability:** Ensures that once a transaction is committed, it will remain so, even in the event of a system failure.
10. **Key-value database and its properties:**
    * **Key-value Database:** Stores data as a collection of key-value pairs, where a key serves as a unique identifier.
    * **Properties:** Fast lookups, simple scalability, and the ability to handle large volumes of data and high throughput

### Part B Answers

#### ****Module 1****

#### Question 11

**Draw an ER diagram to model the application with the following assumptions. Specify key attributes of each entity type and (min, max) constraints on each relationship type**

#### Question 12

**What is the difference between logical data independence and physical data independence? Which one is harder to achieve? Why?**

### ****Question 12a: Data Independence****

**Logical Data Independence**: The ability to change the logical structure of the database without altering the external schemas or application programs. For example, modifying the schema by adding a new attribute or relation should not affect the applications using the database.

**Physical Data Independence**: The ability to modify the physical storage of data without affecting the logical structure visible to applications. For example, changing from sequential to random file storage should be transparent to the application layer.

**Which is Harder to Achieve and Why**: Physical data independence is generally easier to achieve because it deals with the storage and retrieval of data without needing to alter the logical aspects of data that applications interact with. Logical data independence is more challenging because changes in database structure, such as adding new relations or attributes, often require modifications in application logic to accommodate these changes.

### Question 12b: Bank Database ER Diagram Analysis

**i. Strong Entity Types**: Entities that can exist independently, such as Customer, Account, and Branch.

**ii. Weak Entity Type**: Loan might be considered a weak entity if it cannot exist without a specific account or customer. It might have a partial key like a loan number.

**iii. Partial Key and Identifying Relationship**: A partial key might be the loan number, which only makes sense in the context of a customer or account it belongs to. The identifying relationship connects the loan to its customer or account.

**iv. Relationship Constraints**: For example, each customer can have multiple accounts but a maximum of two loans. A branch can have up to 1,000 loans.

**v. Min, Max Constraints Example**: Every customer must have at least one account (min = 1) but no more than two loans (max = 2); a branch can have up to 1,000 loans (max = 1,000).

#### ****Module 2****

#### ****Question 13****

**Part a:**

**Consider the UNIVERSITY database with the following relations:**

**STUDENT (rollNo, name, degree, year, sex, deptNo, advisor)**

**DEPARTMENT (deptId, name, hod, phone)**

**PROFESSOR (empId, name, sex, startYear, deptNo, phone)**

**COURSE (courseId, cname, credits, deptNo)**

**ENROLLMENT (rollNo, courseId, sem, year, grade)**

**TEACHING (empId, courseId, sem, year, classRoom)**

**PREREQUISITE(preReqCourse, courseID)**

**Write relational algebra expressions for the following queries:**

**i. For each department, find its name and the name, sex and phone number of the head**

**of the department.**

**ii. Find courses offered by each department.**

**iii. Find those students who have registered for all courses offered in the department of**

**Computer Science.**

**iv. Obtain the department Ids for departments with no lady professor.**

**v. Obtain the rollNo of girl students who have obtained at least one S grade.**

i. π(name, sex, phone) (σdeptId=hod (DEPARTMENT ⋈ PROFESSOR))

ii. πcname (COURSE ⋈ DEPARTMENT)

iii. STUDENT÷πcourseId​(σdeptNo=’CS’​(COURSE))

iv. π(deptId) (σsex=’F’ (DEPARTMENT ⋈ PROFESSOR))

v. π(rollNo) (σgrade=’S’ and sex=’F’ (STUDENT ⋈ ENROLLMENT))

**Part b: What is a foreign key constraint? Why are such constraints important? What is referential integrity?**

* **Foreign Key Constraint**: Ensures that the value of one or more columns in a table corresponds to one or more columns in another table.
* **Importance**: Maintains the integrity and accuracy of data within the database, ensuring relationships between tables remain consistent.
* **Referential Integrity**: A database concept ensuring that relationships between tables remain consistent, which means no orphan records are present

#### ****Module 3****

### ****Question 14: SQL Queries and DDL Statements****

**Part B:**

#### ****SQL DDL Statements for University Database****

* **Create Table Statements**: Includes the creation of tables with primary and foreign keys to enforce data integrity:

**CREATE TABLE STUDENT (**

**rollNo INT PRIMARY KEY,**

**name VARCHAR(100),**

**degree VARCHAR(50),**

**year INT,**

**sex CHAR(1),**

**deptNo INT,**

**advisor INT,**

**FOREIGN KEY (deptNo) REFERENCES DEPARTMENT(deptId),**

**FOREIGN KEY (advisor) REFERENCES PROFESSOR(empId)**

**);**

**CREATE TABLE DEPARTMENT (**

**deptId INT PRIMARY KEY,**

**name VARCHAR(100),**

**hod INT,**

**phone VARCHAR(15),**

**FOREIGN KEY (hod) REFERENCES PROFESSOR(empId)**

**);**

**CREATE TABLE PROFESSOR (**

**empId INT PRIMARY KEY,**

**name VARCHAR(100),**

**sex CHAR(1),**

**startYear INT,**

**deptNo INT,**

**phone VARCHAR(15),**

**FOREIGN KEY (deptNo) REFERENCES DEPARTMENT(deptId)**

**);**

**Modify Student Table: To add an 'address' attribute to the STUDENT table:**

**ALTER TABLE STUDENT ADD address VARCHAR(255);**

**Delete CS Department: To delete the 'CS' department, considering referential integrity constraints, the SQL might look like:**

**DELETE FROM DEPARTMENT WHERE name = 'CS';**

****Question 15:****

****Part a: What is an assertion? How they differ from triggers?****

****Refer Capsule note: capsule\_notes\_dbms\_module\_3.pdf****

****Question : 7 and 8****

****Part b:****