1. You have entities of ITEM, SOLD-ITEM, SALE and PAYMENT. Which most likely is NOT a relationship?
   a) SALE is paid by PAYMENT
   b) PAYMENT pays for ITEM
   c) ITEM is included in SOLD-ITEM
   d) SALE involves SOLD-ITEM
   e) PAYMENT pays for SALE

2. Which of the following is NOT true about ERDs?
   a) Special symbols are added to show high-level business rules
   b) The diagrams are drawn in a sequential order – from top to bottom
   c) Similar kinds of information are listed together in entities
   d) ERD's are data modeling techniques
   e) Lines are drawn to show relationships among the data

3. Jack is developing an ERD for a small dental practice office patient record system. The dental practice has three dentists, six hygienists, and many patients. A patient is always assigned to the same dentist for all appointments. In particular, he is working on the relationship between dentists and patients. Should it be:
   a) 1 to 1, with a modality of null
   b) 1 to many with a modality of not null
   c) Many to many with a modality of null
   d) Many to many with a modality of not null
   e) 1 to many with a modality of null

4. What is true about creating an entity relationship diagram?
   a) There will be at most seven entities
   b) There will be at most seven relationships
   c) If you identify more than seven entities, analyze and combine until you have seven or less
   d) It is an iterative process
   e) Entities will have at most seven attributes

5. Anthony is working on the cardinality of doctors and patients in a large urban hospital. With the large number of doctors with varying specialties and patients that may have more than one aliment, he thinks the relationship might be noted as:
   a) 1 to 1
   b) 1 to 2
   c) 1 to many
   d) Many to many
   e) Many to 1

6. When normalizing data models, if you take attributes that have multiple values for a single instance of an entity and create separate entities for those attributes you are moving from:
   a) O normal form to 1st normal form (1NF)
   b) 1st normal form (1NF) to 2nd normal form (2NF)
   c) 2nd normal form (2NF) to 3rd normal form (3NF)
   d) Generalized normal form (GNF) to fully normalized form (FNF)
   e) Dependent normal form (DNF) to Independent normal form (INF)

7. A logical data model that does not lead to repeating fields and that the data models leads to tables containing fields that are dependent on the whole identifier is in _____ normal form.
   a) balanced
   b) first

c) primary
d) second
e) third

8. Which of the following can be a multivalued attribute
a) Phone Number
b) Name
c) Date of birth
d) All of the mentioned

9. The attribute name could be structured as an attribute consisting of first name, middle initial, and last name. This type of attribute is called
a) Simple attribute
b) Composite attribute
c) Multivalued attribute
d) Derived attribute

10. If we have 300 and 200 entities in EMP and PRJ entity sets respectively, what would be the maximum number of entities the Handle relationship set would have?

a) 300
b) 200
c) 500
d) 60000

60000

11. Consider the follwing two statements
Statement A: The cardinality of a relation is the number of tuples it contains.
Statement B: The degree of a relation is the number of records it contains.
Which statement(s) is/are correct?

a) Only B
b) A and B
c) Only A
d) Neither A nor B

degree
cardinality

12. Which of the given conditions would the query optimizer most likely decide to execute first (e.g., as one of the bottom-most nodes in the query plan) if it wants to generate an optimal query plan for the SQL query given below? Note: Assume that additional information regarding the statistics of relations R, S and T are not given.

SELECT * FROM R, S, T WHERE R.r = S.s AND S.id = T.id AND R.no = 102 AND S.price > 100;

a) R.r = S.s
b) S.id = T.id
c) S.price > 100
d) **R.no = 102**

13. What is a partial dependency?
a) When a non-key attribute depends on a subset of the primary key
b) When a non-key attribute depends on the entire primary key
c) When a non-key attribute depends on a single attribute
d) When a primary key attribute depends on a non-key attribute
e) When a non-key attribute depends on a foreign key

14. What is transitive dependency?

a) When $A \rightarrow B$ and $B \rightarrow C$, then $A \rightarrow C$

b) When $A \rightarrow B$ and $C \rightarrow B$, then $A \rightarrow C$
c) When $A \rightarrow B$ and $B \leftrightarrow C$, then $A \rightarrow C$
d) When $A \leftrightarrow B$ and $B \rightarrow C$, then $A \rightarrow C$
e) When $A \rightarrow B$ and $B \rightarrow A$, then $A \rightarrow C$

15. What is the difference between a functional dependency and a foreign key?
a) A functional dependency is a relationship between attributes, while a foreign key is a relationship between tables
b) A functional dependency is a relationship between tables, while a foreign key is a relationship between attributes
c) A functional dependency is used for data integrity, while a foreign key is used for data redundancy
d) A functional dependency is used for data redundancy, while a foreign key is used for data integrity
e) A functional dependency is used for data scalability, while a foreign key is used for data security

16. What is the purpose of functional dependencies in database design?
a) To ensure data redundancy
b) To ensure data integrity
c) To improve data scalability
d) To enhance data security

17. How do functional dependencies help in data modeling?
a) To ensure data redundancy
b) To ensure data integrity
c) To improve data scalability
d) To enhance data security
e) To reduce data consistency

18. Consider the relation Schema R (A,B,C). If $A \rightarrow B$ and $B \rightarrow C$ are functional dependencies on R, then which of the following can be considered as a candidate?
a) B
b) AB
c) BC
d) C
e) A

19. Let $r_i(z)$ and $w_i(z)$ denote read and write operations respectively on a data item z by a transaction T_i. Consider the following two Schedules
S_1 : $r_1(x)$ $r_1(y)$ $r_2(x)$ $r_2(y)$ $w_2(y)$ $w_1(x)$
S_2 : $r_1(x)$ $r_2(x)$ $r_2(y)$ $w_2(y)$ $r_1(y)$ $w_1(x)$
Which of the following options are correct?
a) S_1 is conflict serializable, S_2 is not conflict serializable
b) Both S_1 and S_2 are conflict serializable
c) S_1 is not conflict serializable, S_2 is conflict serializable
d) Neither S_1 nor S_2 is conflict serializable
e) None of the above

20. A many-to-one relationship exists between entity sets r_1 and r_2. How will it be represented using functional dependencies if Pk(r) denotes the primary key attributes of relation r?

a) $PK(r_2) \rightarrow PK(r_1)$
b) $PK(r_2) \rightarrow PK(r_1)$ and $PK(r_1) \rightarrow PK(r_2)$
c) $PK(r_1) \rightarrow PK(r_2)$
d) $PK(r_2) \rightarrow PK(r_1)$ or $PK(r_1) \rightarrow PK(r_2)$
e) Can be A or B

21. Let R(A, B, C, D) be a relation schema and $F = \{A \rightarrow BC, AB \rightarrow D, B \rightarrow C\}$ be the set of functional dependencies defined over R. Which of the following represents the closure of the attribute set {B}?
a) {A , C, D}
b) {A, B, C}
c) {B, C}
d) {B}
e) None of the above

22. Let S be the following schedule of operations of three transactions T1, T2 and T3 in a relational database system: R2(Y), R1(X), R3(Z), R1(Y), W1(X), R2(Z), W2(Y), R3(X), W3(Z), Consider the statements P and Q below:
P: S is conflict-serializable.
Q: If T3 commits before T1 finishes, then S is recoverable.
Which one of the following choices is correct?
a) Both P and Q are true
b) P is false and Q is true
c) Both P and Q are false
d) P is true and Q is false
e) None of the above

23. *I* and *J* are _____ if they are operations by different transactions on the same data item, and at least one of them is a write operation.
a) Overwriting
b) Isolated
c) Durable
d) Conflicting
e) Atomic

24. Consider the following Transaction

```
Transaction
.....
Commit;
Rollback;
```

What will Rollback do in this case?

a) Undo the transactions before commit
b) Clears all transactions
c) Redo the transactions before commit
d) No action

e) Will Rollback the committed transaction

25. What is TRUE about Isolation?
    a) By using the data used during a transaction, the second transaction will not be able to use it until the first has been executed.
    b) The data item X cannot be accessed by any other transaction T2 until the transaction T1 is completed and the data item X is used by the transaction T1.
    c) It enforced the isolation property via its concurrency control subsystem.
    d) All of the above
    e) None of the above

26. Consider the following set of functional dependencies on the scheme (A, B, C)
    The canonical cover for this set is

    A→BC
    B→C
    A→B
    AB→C

    a) A→BC and B→C
    b) A→BC and AB→C
    c) A→BC and A→B
    d) A→B and B→C
    e) None of the above

27. Given an instance of the relation R(ABCD)

A	B	C	D
A1	B1	C1	D1
A1	B2	C2	D2
A2	b2	C2	D3
A3	B3	C4	D3

Which of the following functional dependencies hold?

a) {AB} → D and D → A
b) {AB} → C and B → D
c) {AB} → D and A → D
d) {AB} → C and B → C
e) None of the above

Consider following case study to answer Q#28 to Q#34

Case Study: E-commerce Platform

An e-commerce platform aims to develop a database system to manage information related to products, orders, and customers. They have provided the following requirements:

Products:

Each product has a unique identification number, name, category, and price.
Products can belong to multiple categories.
Each product can be associated with multiple orders.

Customers:

Each customer has a unique identification number, name, email, and address.
Customers can place multiple orders.
Each customer can be associated with multiple products through orders.

Orders:
Each order has a unique identification number, date, and status (e.g., processing, shipped).
An order can contain multiple products.
Each order is associated with a single customer.

Categories:
Each category has a unique identification number and name.
Products can belong to multiple categories.
Each category can have multiple products.

28. Which of the following attributes would be part of the "Order" entity?
    a) Product ID
    b) Customer Name
    c) Order Status
    d) Product Category

29. Which relationship type represents the association between an order and its customer?
    a. One-to-One
    b. One-to-Many
    c. Many-to-One
    d. Many-to-Many

30. Which SQL query can be used to retrieve the names of customers along with their corresponding orders' status?
    a) SELECT Customers.name, Orders.status FROM Customers JOIN Orders ON Customers.customer_id = Orders.customer_id;
    b) SELECT Customers.name, Orders.status FROM Customers, Orders WHERE Customers.customer_id = Orders.customer_id;
    c) SELECT Customers.name, Orders.status FROM Customers LEFT JOIN Orders ON Customers.customer_id = Orders.customer_id;
    d) SELECT Customers.name, Orders.status FROM Customers RIGHT JOIN Orders ON Customers.customer_id = Orders.customer_id;
    e)

31. Which SQL query can be used to retrieve the total number of products sold in each category?
    a) SELECT Categories.name, COUNT(Products.product_id) FROM Categories JOIN Products ON Categories.category_id = Products.category_id GROUP BY Categories.name;
    b) SELECT Categories.name, COUNT(DISTINCT Products.product_id) FROM Categories JOIN Products ON Categories.category_id = Products.category_id GROUP BY Categories.name;
    c) SELECT Categories.name, SUM(Products.quantity_sold) FROM Categories JOIN Products ON Categories.category_id = Products.category_id GROUP BY Categories.name;
    d) SELECT Categories.name, SUM(Products.product_id) FROM Categories JOIN Products ON Categories.category_id = Products.category_id GROUP BY Categories.name;

2. Which SQL query can be used to retrieve the customers who have placed orders with a total value exceeding $500?
    a) SELECT Customers.* FROM Customers JOIN Orders ON Customers.customer_id = Orders.customer_id JOIN Products ON Orders.order_id = Products.order_id WHERE SUM(Products.price) > 500;
    b) SELECT Customers.* FROM Customers JOIN Orders ON Customers.customer_id = Orders.customer_id JOIN Products ON Orders.order_id = Products.order_id GROUP BY Customers.customer_id HAVING SUM(Products.price) > 500;

c) SELECT Customers.* FROM Customers JOIN Orders ON Customers.customer_id = Orders.customer_id JOIN Products ON Orders.order_id = Products.order_id WHERE Products.price > 500;
d) SELECT Customers.* FROM Customers JOIN Orders ON Customers.customer_id = Orders.customer_id WHERE SUM(Orders.total_value) > 500;

33. Which SQL query can be used to retrieve the products that have been ordered by more than five customers?
a) SELECT Products.* FROM Products JOIN Orders ON Products.product_id = Orders.product_id GROUP BY Products.product_id HAVING COUNT(DISTINCT Orders.customer_id) > 5;
b) SELECT Products.* FROM Products JOIN Orders ON Products.product_id = Orders.product_id WHERE COUNT(Orders.customer_id) > 5;
c) SELECT Products.* FROM Products JOIN Orders ON Products.product_id = Orders.product_id GROUP BY Products.product_id HAVING COUNT(Orders.customer_id) > 5;
d) SELECT Products.* FROM Products JOIN Orders ON Products.product_id = Orders.product_id WHERE COUNT(DISTINCT Orders.customer_id) > 5;

34. Which SQL query can be used to retrieve the customers who have not placed any orders yet?
a) SELECT Customers.* FROM Customers JOIN Orders ON Customers.customer_id = Orders.customer_id WHERE Orders.order_id IS NULL;
b) SELECT Customers.* FROM Customers LEFT JOIN Orders ON Customers.customer_id = Orders.customer_id WHERE Orders.order_id IS NULL;
c) SELECT Customers.* FROM Customers JOIN Orders ON Customers.customer_id = Orders.customer_id WHERE Orders.order_id = NULL;
d) SELECT Customers.* FROM Customers RIGHT JOIN Orders ON Customers.customer_id = Orders.customer_id WHERE Orders.order_id = NULL;

35. A _____ on an attribute of a relation is a data structure that allows the database system to find those tuples in the relation that have a specified value for that attribute efficiently, without scanning through all the tuples of the relation.
a) Index
b) Reference
c) Assertion
d) Timestamp

36. We can test for the nonexistence of tuples in a subquery by using the _____ construct.
a) Not exist
b) Not exists
c) Exists
d) Exist

37. Database _____ which is the logical design of the database, and the database _____ which is a snapshot of the data in the database at a given instant in time.
a) Instance, Schema
b) Relation, Schema
c) Relation, Domain
d) Schema, Instance

43. Which of the following statements is/are true abouf surrogate key?
a) Surrogate keys cannot be used as primary keys.
b) Surrogate keys can be used as primary keys and is auto-generated numeric column in the table.
c) If any relation has more than one candidate key, then after choosing the primary key from those candidate keys, rest of candidate keys are known as surrogate keys of that table.
d) If a primary key of a table consists of more than one column then the primary key is also called a surrogate key.
e) All of the above.

44. Suppose that two relations R(A,B) and S(A,B) have exactly the same schema and consider the following equalities.

i) $R \cap S = R - (R - S)$
ii) $R \cap S = S - (S - R)$
iii) $R \cap S = (R \cup S) - ((R - S) \cup (S - R))$
iv) $R \cap S = R * S$ // * is natural join

Which of the above equalities hold in relational algebra?
a) (I) and (III) only.
b) (I) and (II) only.
c) (I), (II) and (III) only.
d) (II), (III) and (IV) only.
e) All

45. Which of the following relational algebra expression(s) would list the student id (sid) and name (sname) of each student along with the course id (cid) and name of the courses (cname) for which the student has registered? Students who are not registered for any course must also be listed.

a) $\pi_{\text{sid, sname, cid, cname}}$((Student $\bowtie_{\text{sid=sid}}$ Registered) * Course)

b) $\pi_{\text{sid, sname, cid, cname}}$ ((Student $\bowtie_{\text{sid=sid}}$ Registered) * Course)

c) $\pi_{\text{sid, sname, cid, cname}}$ ((Student $\bowtie_{\text{sid=sid}}$ Registered) * Course)

d) $\pi_{\text{sid, sname, cid, cname}}$((Student * Registered) * Course)

46. The statements given below are associated with Referential Integrity Constraints.
(i) A foreign key attribute can never contain a null value.
(ii) A foreign key attribute is a record that always refers to another record which does not contain null.
(iii)Referential integrity constraint is specified between two relations and is used to maintain the consistency.
Which of the above is/are true?

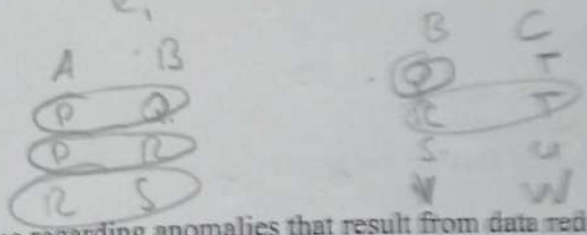
a) (i) only
b) (ii) only
c) (iii) only
d) (i) and (iii)only
e) (ii) and (iii)only

47. There needs to be which of the following conditions for each nontrivial dependency of function X on function Y for a relation to be in third normal form.

a) A super key is X.
b) Every element of Y is a part of some candidate key, i.e, Y is a prime attribute.
c) Either A or B
d) None of the above

48. Suppose that the relation R1 (A, B) has tuples {(P, Q), (P, R), (R, S)}, and the relation R2 (B, C) has tuples {(Q, T), (R, T), (S, U), (V, W)}.
Consider the following SQL query.
SELECT * FROM R1 LEFT OUTER JOIN R2 ON R1.B = R2.B;
What is the number of tuples in the result of the above SQL query?

a)2
b)3
c)4
d)5
e)6



49. Which of the following are true regarding anomalies that result from data redundancy?
a) Only creation and update anomalies are resulted from data redundancy.
b) Only insertion, update and creation anomalies are resulted from data redundancy.
c) Only insertion, update and delete anomalies are resulted from data redundancy.
d) Only deletion anomalies are resulted from data redundancy.
e) Only insertion and update anomalies are resulted from data redundancy.

50. Which of the following sequence of operations would find the ids of the employees (Eid) who get more than Rs. 20,000 salary and who have worked more than 5 years on project 5?

a) πEid(σPid=5 (σSalary>20000(Employee) $\bowtie$ Eid=Emp_Id σDuration>5 (Involves)))
b) πEid (σPid=5 (σSalary>20000 (Employee)) $\bowtie$ Eid=Emp_IdEmp_Id (σDuration>5 (Involves)))
c) πEid (σSalary>20000 (Employee)) $\cap$ πEmp_Id (σDuration>5 and Pid=5 (Involves))
d) πEid(σPid=5 (σSalary>20000 and Duration>5 (Employee $\cap$ Involves)))
e) All of above.

Final Exam-D_2024-05-23_key

Q No	Correct
Final Exam - Section1	
1	B
2	B
3	B
4	D
5	D
6	A
7	D
8	A
9	B
10	D
11	C
12	D
13	A
14	A
15	A
16	B
17	B
18	B
19	C
20	C
21	C
22	D
23	D
24	D
25	D
26	D
27	D
28	C
29	C
30	C
31	A
32	B
33	C
34	B
35	A
36	B
37	D
38	B
39	D
40	C
41	C
42	B
43	B
44	E
45	A
46	B
47	C
48	B
49	C
50	C

Question 1 (20 marks)
Consider the following relational Schema given below.

PART A

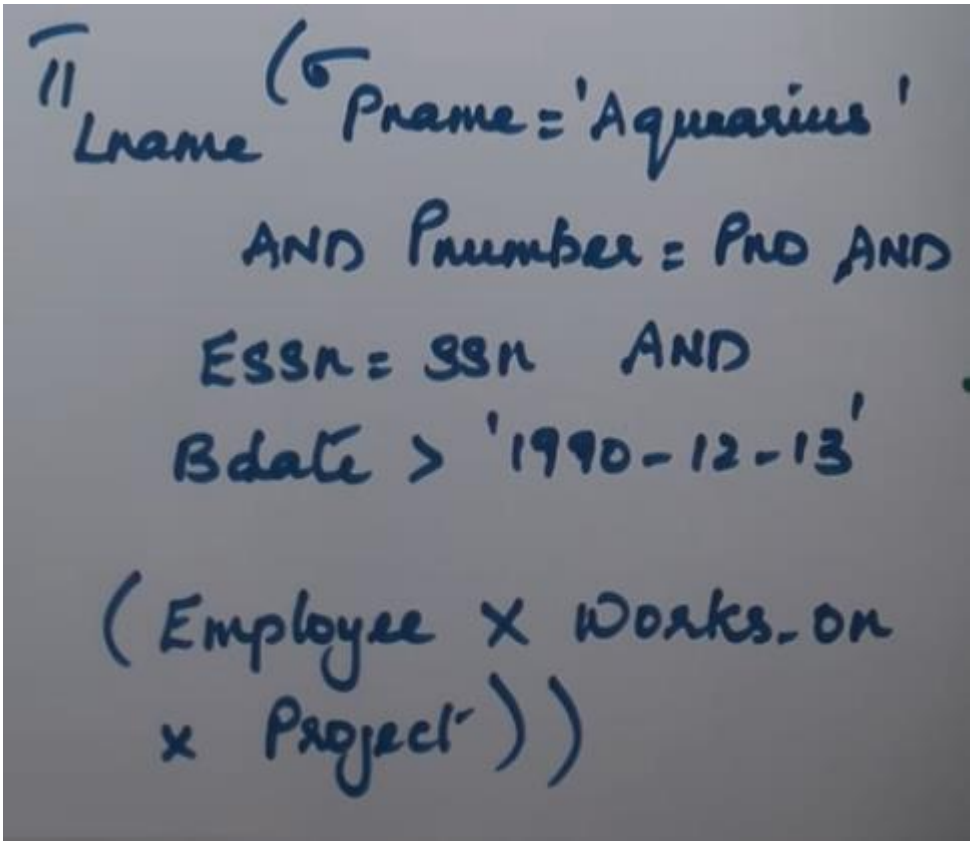
EMPLOYEE(Fname,Lname,SSN, Bdate , Address,Gender, Salary,SuperSSN,Dno)
WORKS_ON(ESSN,Pno, Hours)
PROJECT(Pname,Pnumber, PLocation, Dnum)

You have to write Relational Algebric expression against this query and also apply heuristic optimization to generate query tree for the query given below.

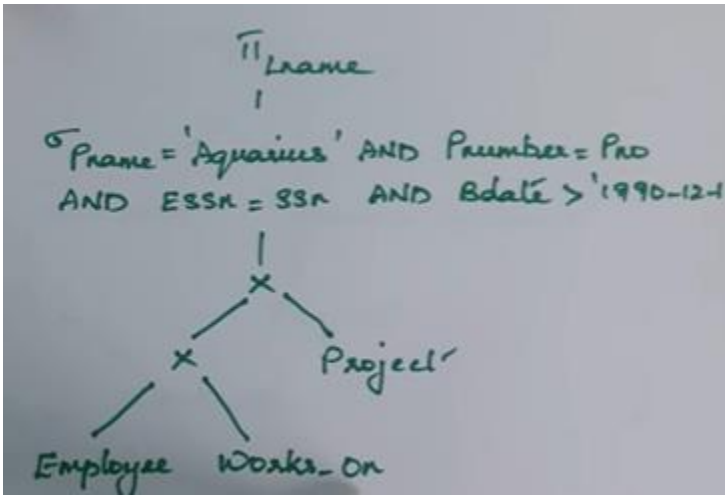
SQL:

```
SELECT Lname
 FROM EMPLOYEE, WORKS_ON, PROJECT
 WHERE Pname='Aquarius'
AND Pnumber =Pno
AND ESSN=SSN
AND Bdate>  '1990-12-13''
```

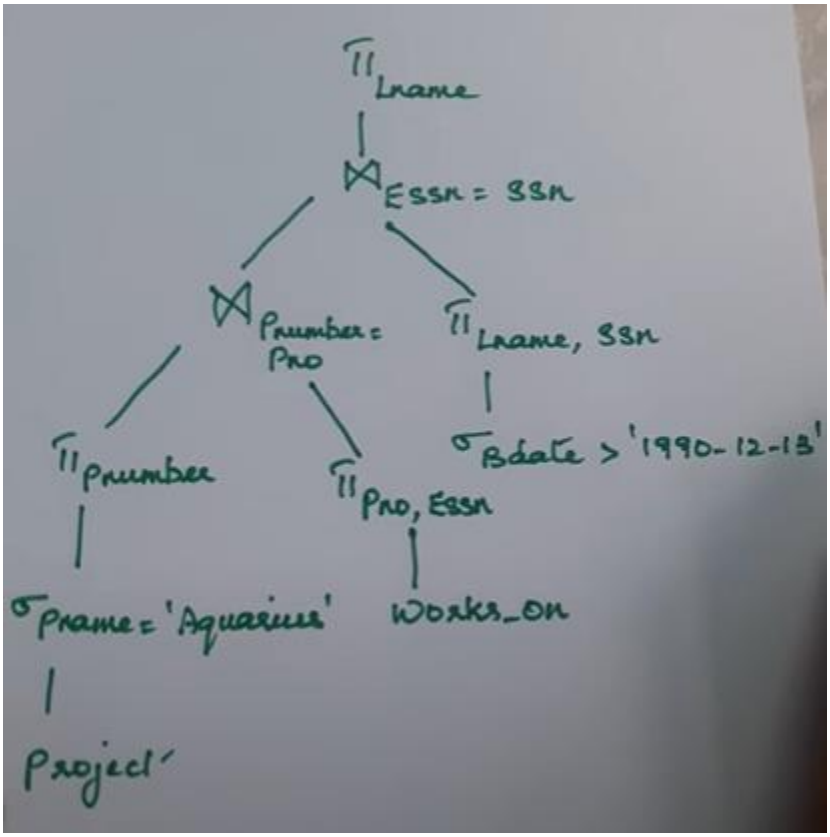
Relational Algebra: [2]



Basic Query Tree: [3]



Optimized Query tree [5]



Part B

Consider the following SQL Query and generate query tree

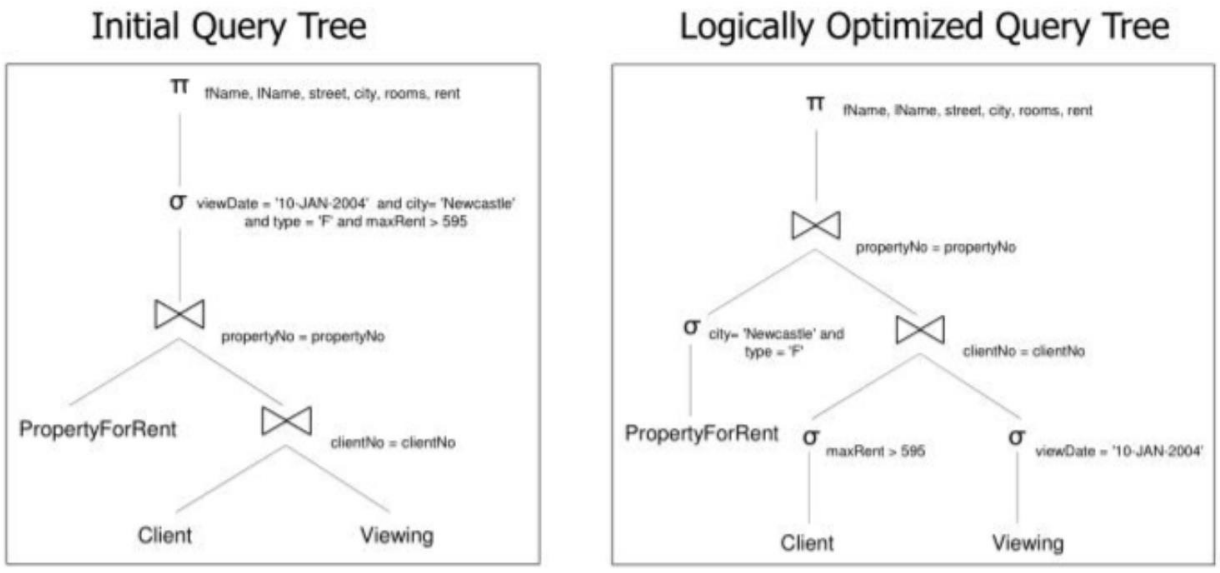
SELECT C.fName, C.lName, P.street, P.city, P.rooms, P.rent

```
 FROM PropertyForRent P, Client C, Viewing V
 WHERE P.propertyNo = V.propertyNo
 AND V.clientNo = C.clientNo
AND V.viewDate = '10-JAN-2004'
 AND P.city = 'Newcastle'
AND P.type = 'F'
AND C.maxRent > 595
```

Basic Query Tree:

Optimized Query tree

Answer



Question 2 (10 Marks)
Suppose we have a query to retrieve the students with age 18 and studying in class DESIGN_01. We can get all the student details from STUDENT table, and class details from CLASS table. We can write it in two different waysConsider the following two Relational algebraic form of both queries. Create a tree diagram for each query, and then analyze and justify which query is more efficient in terms of execution time and resource utilization.

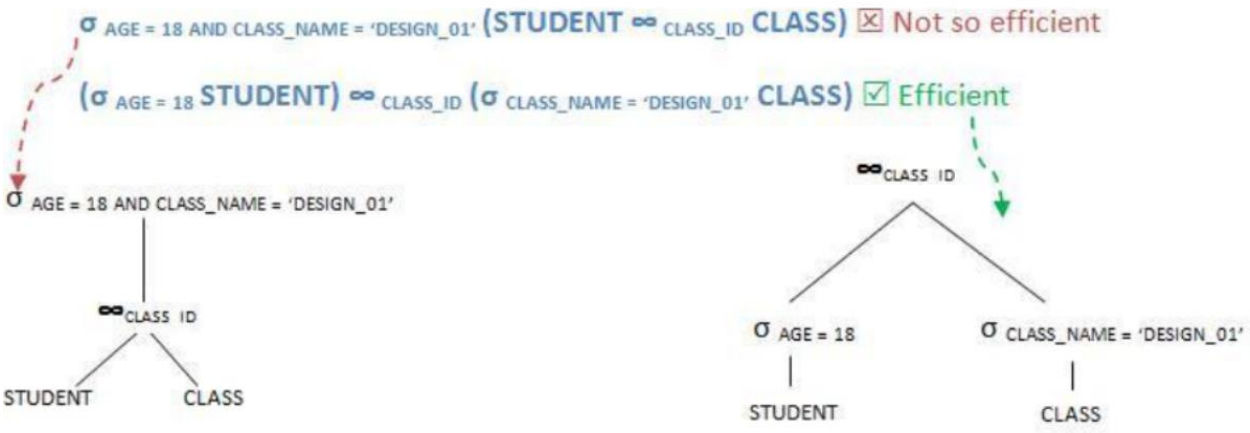
Query 1:

$$\sigma_{Age=18 \text{ AND } CLASS\_NAME='DESIGN+01}(STUDENT \bowtie_{CLASS\_ID} CLASS)$$

Query 2:

$(\sigma_{Age=18}\ STUDENT) \bowtie_{CLASS\_ID} (\sigma_{CLASS\_NAME='DESIGN+01'}\ CLASS)$

Answer:



heck whether the given schedule S is conflict serializable or not? List all the conflicting operations and Draw precedence graph- [Marks 5+5]

S : $R_1(A)$, $R_2(A)$, $R_1(B)$, $R_2(B)$, $R_3(B)$, $W_1(A)$, $W_2(B)$

Solution-

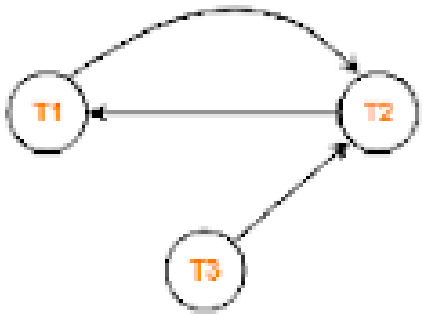
Step-01:

List all the conflicting operations and determine the dependency between the transactions-

- $R_2(A)$, $W_1(A)$ $(T_2 \rightarrow T_1)$
- $R_1(B)$, $W_2(B)$ $(T_1 \rightarrow T_2)$
- $R_3(B)$, $W_2(B)$ $(T_3 \rightarrow T_2)$

Step-02:

Draw the precedence graph-



- Clearly, there exists a cycle in the precedence graph.
- Therefore, the given schedule S is not conflict serializable.

Question 3:- Consider two serial transactions T1 and T2. Make these two transactions concurrent so that it becomes serializable. Check whether the given schedule and the one you found are conflict equivalent or not? [Marks 5+5]

Let A=50, B=60, N=5,M=10

T1	T2
Read_item(A); A:=A-N; Write_item(A); Read_item(B); B:=B+N; Write_item(B);	
	Read_item(A); A:=A+M; Write_item(A);

Solution:

T1	T2
Read_item(A); A:=A-N; Write_item(A);	
	Read_item(A); A:=A+M; Write_item(A);
Read_item(B); B:=B+N; Write_item(B);	

Checking Conflict Equivalence
Given: A= 55
B= 65
Found: A = 55
B = 65

Question 4:- Consider two Transactions T1 and T2 as below. Run operations in both transactions in parallel. How will you explain the temporary update problem and How it suffers Transactions? Which ACID property solves the Temporary Update Problem and How?
[Marks 5+5]

T1	T2

Solution:-

T1	T2
Read_item(A); A:=A-N; Write_item(A);	
	Read_item(A); A:=A+M; Write_item(A);
Read_item(B); **T1 Fails**	

The Acid Property that solves the temporary update problem is **"isolation"** because it doesn't show the updates of the transactions until committed.

Q5: Transaction Processing **[6+4=10 marks]**

Provide two Transactions T1 and T2 below. Run operations in both transactions in parallel. How will you explain the temporary update problem and How it suffers Transactions? Which ACID property solves the Temporary Update Problem and How?

T1	T2

Q6: Relational Algebra **[15x2=30 marks]**

Given the following set of table definitions:

Book (Book Id, ISBN, Title, Publisher, Year)
Student (Student Id, Student_Name, Course_Name, Age)
Author (Author Name, Age, Email)
Borrow (Book Id*, Student Id*, Borrowed on)
Wrote (Book Id*, Author Name*)
Classification (Book Id*, Genre_Name*)
Genre (Genre_Name, Description)

Produce the complete relational algebraic expression (NOT SQL) for the following queries by replacing the ? Symbol given in each statements. Also don't try to change the position of any relation or operators.

1.Project the names and ages of students who borrowed books published after 2020.

$\pi_{\text{Student.Student\_Name, Student.Age}}(Student \bowtie Borrow \bowtie (\sigma_{\text{Book.Year}>2010}(Book)))$

1.Student
2. Borrow
3.Book.Year
4. Book

2. Determine all books that are both classified as 'Fiction' and 'Thriller' genres.

$\sigma_{\text{Genre.Genre\_Name='Fiction'}}(Classification \bowtie Genre)) \cap (\sigma_{\text{Genre.Genre\_Name='Thriller'}}(Classification \bowtie Genre)) \bowtie Book$

1.Classification
2. ∧
3.σGenre.Genre_Name='Thriller'
4.Genre

3. Find all authors who have written books in multiple genres.

$\pi_{\text{Author.Author\_Name}}(\sigma_{\text{count(Genre.Genre\_Name)}>1}(Wrote \bowtie Classification \bowtie Genre \bowtie Author))$

1.πAuthor.Author_Name
2.Genre.Genre_Name
3.Classification
4.Author

4. *Determine all authors who have not written any books classified under the 'Non-Fiction' genre.*

$\pi_{\text{Author.Author\_Name}}(Author - (\sigma_{\text{Genre.Genre\_Name='Non-Fiction'}}(Wrote \bowtie Classification \bowtie Genre \bowtie Author)))$

1.πAuthor.Author_Name
2. -
3.Classification
4. Author

5. *Determine all students who have borrowed books but are not enrolled in any course.*

$\pi_{\text{Student.Student\_Name}}((Student \bowtie Borrow) - (Student \bowtie \text{Course}))$

1.Student_Name
2. Student
3. -
4.Course

6. *Find the union of books borrowed by students aged 25 or younger and books borrowed by students aged 30 or older.*

$(\sigma_{\text{Student.Age} \leq 25}(Student \bowtie Borrow \bowtie Book)) \cup (\sigma_{\text{Student.Age} \geq 30}(Student \bowtie Borrow \bowtie Book))$

1.Student.Age≤25
2.Borrow
3.Student.Age≥30
4.Student

7. *Select all students who are not enrolled in any course.*

$\pi_{\text{Student.Student\_Name}}(\sigma_{\text{Student.Course\_Name=NULL}}(Student))$

1.Student.Student_Name
2. Student.Course_Name
3. Null
4. Student

8. *Find the difference between all students and students who have borrowed books.*

$\pi_{\text{Student.Student\_Name}}(Student) - \pi_{\text{Student.Student\_Name}}(Borrow \bowtie Student)$

1.Student.Student_Name
2. Student.Student_Name
3. Borrow
4. Book

9. Retrieve the titles of books authored by 'John Doe'.

πBook.Title(σWrote.Author_Name='John Doe'($Book \bowtie$ Wrote))

1.Book.Title
2.Wrote.Author_Name
3. Book
4. Wrote

10. Find all books borrowed by students along with the corresponding student names.

πBook.Title, Student.Student_Name($Book \bowtie$ Borrow $\bowtie$ Student)

1.Book.Title
2. Book
3.Borrow
4.Student

11. List all books of the 'Mystery' genre borrowed after January 1, 2023.

πBook.Title(σGenre.Genre_Name='Mystery'$\wedge$Borrow.Borrowed_on>'2023-01-01'($Book \bowtie$ Classification $\bowtie$ Genre $\bowtie$ Borrow))

1.σGenre.Genre_Name
2. ∧
3.'2023-01-01'
4. Classification

12. List the titles of books borrowed by students enrolled in courses related to 'Engineering' or 'Computer Science':

πBook.Title(($Student \bowtie Borrow \bowtie Book$) $\bowtie$ (σStudent.Course_Name='Engineering'$\vee$Student.Course_Name='Computer Science'($Student$)))

1.πBook.Title
2.Borrow
3. ∨
4. Student

13. List all books borrowed by students enrolled in the 'Computer Science' course and not borrowed by students in the 'History' course.

πBook.Title(($Book \bowtie Borrow \bowtie Student$) $\bowtie \sigma$Student.Course_Name='Computer Science'($Student$) − ($Book \bowtie Borrow \bowtie \sigma$Student.Course_Name='History'($Student$)))

1.πBook.Title

2. Student.Course_Name

3. -

4. Borrow

14. *Retrieve the names of authors who have written books classified under both 'Fiction' and 'Mystery' genres.*

$\pi_{\text{Author.Author\_Name}}((Wrote \bowtie Classification \bowtie Genre) \div (\sigma_{\text{Genre.Genre\_Name='Fiction'}}(Classification \bowtie Genre) \cup \sigma_{\text{Genre.Genre\_Name='Mystery'}}(Classification \bowtie Genre)))$

1.Classification

2. ÷

3. ∪

4. Genre

15. *Find all books that have not been borrowed by any student under the age of 20.*

$\pi_{\text{Book.Title}}(Book - \pi_{\text{Book.Book\_Id}}(\sigma_{\text{Student.Age<20}}(Student \bowtie Borrow \bowtie Book)))$

1.-

2. Student.Age<20,

3. Student

4. Borrow

Question 7:

a. **Find the flight details of passengers who have booked more than two flight.**

```
SELECT p.FirstName, p.LastName, f.FlightNumber, f.Origin,
f.Destination
FROM Passengers p
JOIN Bookings b ON ______1_____________
JOIN _____2_____ ON ________3_____________
WHERE _____4______ IN (
    SELECT PassengerID
    FROM Bookings
    GROUP BY PassengerID
    HAVING COUNT(___5___) > 2
);
```

1. p.PassengerID = b.PassengerID
2. Flights f
3. b.FlightID = f.FlightID
4. p.PassengerID
5. BookingID

b. Find the average booking count per flight for each origin airport.

```
SELECT fb.Origin, ______1_____ AS AvgBookings
FROM (
    SELECT f.FlightID, f.Origin, ______2____ AS BookingCount
    FROM Flights f
    ____________________________3____________________________
    GROUP BY _____4______
) fb
GROUP BY _____5______;
```

1. AVG(BookingCount)
2. COUNT(b.BookingID)
3. LEFT JOIN Bookings b ON f.FlightID = b.FlightID
4. f.FlightID
5. fb.Origin

c. List flights with their respective pilots and co-pilots names (First and Last Name).

```
SELECT f.FlightNumber,
       c1.FirstName || ' ' || c1.LastName AS Pilot,
       c2.FirstName || ' ' || c2.LastName AS CoPilot
FROM _____1________f
JOIN ___________2_______________ AND _________3__________
JOIN ___________4_______________ AND _________5__________;
```

1. Flights
2. Crew c1 ON f.FlightID = c1.FlightID
3. c1.Role = 'Pilot'
4. Crew c2 ON f.FlightID = c2.FlightID
5. c2.Role = 'Co-Pilot'

d. List the crew members who have worked on the highest number of different flights.

```
SELECT c.FirstName, c.LastName, ____1___ AS MaxFlights
FROM (
    SELECT _____2____, _____3_______ AS FlightCount
    FROM Crew
    GROUP BY CrewID
) fc
JOIN Crew c ON _____________4_____________
```

```
GROUP BY c.CrewID
ORDER BY ____5_____ DESC
LIMIT 1;
1. MAX(FlightCount)
2. CrewID
3. COUNT(FlightID)
4. c.CrewID = fc.CrewID
5. MaxFlights
```

e. Retrieve flights with more bookings than the average number of bookings across all flights.

```
SELECT f.FlightNumber, _______1_______ AS TotalBookings
FROM Flights f
LEFT JOIN Bookings b ON f.FlightID = b.FlightID
GROUP BY f.FlightID
HAVING _______2______ > (
    SELECT AVG(BookingCount)
    FROM (
        ___________3___________ AS BookingCount
        ___________4___________
        ___________5___________
    ) AS AvgBookings
);

1. COUNT(b.BookingID)
2. TotalBookings
3. SELECT COUNT(BookingID)
4. FROM Bookings
5. GROUP BY FlightID
```

f. List flights and their passengers who have booked the same seat.

```
SELECT f.FlightNumber, b.SeatNumber, GROUP_CONCAT(p.FirstName || ' '
|| p.LastName) AS Passengers
FROM Flights f
JOIN ___________________1_____________________
JOIN Passengers p ON ____________2_____________
WHERE b.SeatNumber IN (
    SELECT SeatNumber
    FROM Bookings
    GROUP BY _________3_________
    ___________4_____________
)
```

_____5_____;

```
1. Bookings b ON f.FlightID = b.FlightID
2. b.PassengerID = p.PassengerID
3. SeatNumber
4. HAVING COUNT(*) > 1
5. GROUP BY f.FlightNumber, b.SeatNumber;
```