KIIT Deemed to be University
Online End Semester Examination(Autumn Semester-2021)

Subject Name & Code: Database Management Systems (CS 2004)

Applicable to Courses: B.Tech (CSSE & CSCE)

Full Marks=50 **Time:2 Hours**

SECTION-A(Answer All Questions. Each question carries 2 Marks)

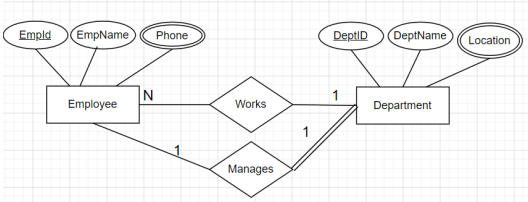
Time:30 Minutes **(7×2=14 Marks)**

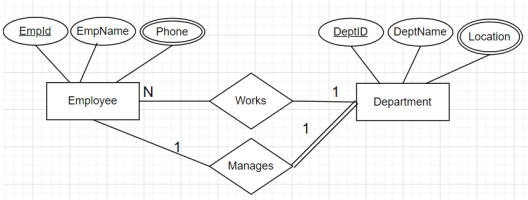
<table>
<tr><th>Question No</th><th>Question Type (MCQ/SAT)</th><th>Question</th><th>CO Mapping</th><th>Answer Key (For MCQ Questions only)</th></tr>
<tr><td>Q.No:1</td><td>MCQ</td><td>Which one correct?
a) Primary Key ⊂ Super Key ⊂ Candidate Key
b) Primary Key ⊂ Candidate Key ⊂ Super Key
c) Candidate Key ⊂ Primary Key ⊂ Super Key
d) Super Key ⊂ Candidate Key ⊂ Primary Key</td><td>CO4</td><td>B</td></tr>
<tr><td></td><td>MCQ</td><td>The following table has two attributes A and C where A is the primary key and C is the foreign key referencing A with ON DELETE CASCADE.
<table>
<tr><th>A</th><th>C</th></tr>
<tr><td>2</td><td>4</td></tr>
<tr><td>3</td><td>4</td></tr>
<tr><td>4</td><td>3</td></tr>
<tr><td>5</td><td>2</td></tr>
<tr><td>7</td><td>2</td></tr>
<tr><td>9</td><td>5</td></tr>
<tr><td>6</td><td>4</td></tr>
</table>
What is the set of all tuples that must be additionally deleted to preserve referential integrity when the tuple (2,4) is deleted?
a) (5,2) (7,2)
b) (4,3)</td><td>CO3</td><td>A</td></tr>
</table>

<table>
<tr><td></td><td></td><td>c) (3,4) (6,4)
d) None of these is correct</td><td></td><td></td></tr>
<tr><td></td><td>MCQ</td><td>Let R (A, B, C) be a relation as follows:
<table><tr><td>A</td><td>B</td><td>C</td></tr><tr><td>10</td><td>20</td><td>30</td></tr><tr><td>70</td><td>20</td><td>30</td></tr><tr><td>10</td><td>20</td><td>50</td></tr></table>What is the correct output of the following SQL query?

SELECT COUNT(*) FROM (SELECT r1.A, r1.B, r1.C FROM R r1, R r2 WHERE r1.B = r2.B AND r1.C = r2.C);

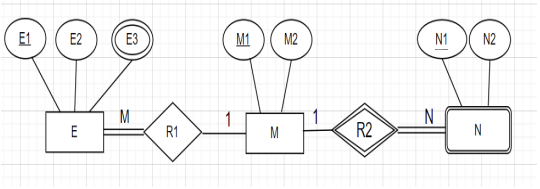
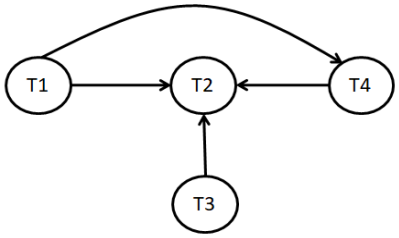
(a) 8
(b) 7
(c) 6
(d) 5</td><td>CO3</td><td>D</td></tr>
<tr><td></td><td>MCQ</td><td>‘_ _ _ %’ matched any string of

a) At least three characters
b) At most three characters
c) Exactly three characters
d) Exactly three characters ending with %</td><td>CO3</td><td>A</td></tr>
<tr><td><u>Q.No:2</u></td><td>MCQ</td><td>Consider the ER diagram as follows:
 

How many foreign keys can be created while mapping to tables from the given ER Model?
(a) 3
(b) 4
(c) 5
(d) 6</td><td>CO2</td><td>B</td></tr>
<tr><td></td><td>MCQ</td><td>Consider the ER diagram as follows:
 

How tables can be created in this example?

(a) 2
(b) 3
(c) 4
(d) 5</td><td>CO2</td><td>C</td></tr>
</table>

	MCQ	Let E1 and E2 be two entities in an ER diagram with simple single valued attributes (with one of the attribute serve as a primary key). R1 and R2 are two relationships between E1 and E2, where R1 is one-to-many and R2 is many-to-many. R1 and R2 do not have any attributes of their own. What is the minimum number of tables and foreign keys are required to represent this situation in relational model? a) (2, 2) b) (2, 3) c) (3, 2) d) (3, 3)	CO2	D
	MCQ	 Which of the following is a correct attribute set of one of the tables for the correct answer to the above question? [Here, N is a weak entity set]. a) {E1, E2,E3} b) {E1, E2, M1} c) {M1, M2, E1} d) {N1,N2, M1, M2}	Co2	B
Q.No:3	MCQ	 Consider the above precedence graph of concurrent execution of four transactions, which is conflict serializable. Which of the following is the correct corresponding serial schedule of the above graph? a) {T1→T4→T2→T3} b) {T3→T2→T1→T4} c) {T3→T1→T4→T2} d) {T2→T1→T3→T4}	CO5	C
		Consider the following concurrent schedule: S : R1(A) , R2(A) , R2(B) , R3(B) , W1(A) , W2(B). The above schedule is I. Conflict serializable schedule. II. Not conflict serializable schedule. III. Conflict equivalent to {T1→T2→T3} IV. Conflict equivalent to		

<table>
<tr><td></td><td>MCQ</td><td>{T3→T2→T1}

Which of the following is true?
a) Only I
b) Only II
c) I and III
d) I and IV</td><td>CO5</td><td>D</td></tr>
<tr><td></td><td>MCQ</td><td>The Timestamp Ordering Protocol is used to order the transactions based on their Timestamps. Let the TS(T1) = 3 and TS(T2) =4. Then, which among the following is incorrect about Timestamp Ordering Protocol.
I. The transaction T1 issuing a Read (X) operation will be rejected if TS(T1) < W_TS (X).
II. This protocol is free from deadlock thus no transaction ever waits.
III. The transaction T1 issuing Write(X) operation will be executed if TS(T1) < R_TS(X).

a) Only I
b) Only II
c) Only III
d) Both I and III</td><td>CO5</td><td>C</td></tr>
<tr><td></td><td>MCQ</td><td>Consider the following schedule with locking:
<table><tr><th>T1</th><th>T2</th></tr><tr><td>Lock-X(A)</td><td></td></tr><tr><td>R(A)</td><td></td></tr><tr><td>W(A)</td><td></td></tr><tr><td></td><td>Lock-X(B)</td></tr><tr><td></td><td>R(B)</td></tr><tr><td></td><td>W(B)</td></tr><tr><td></td><td>Lock-X(A)</td></tr><tr><td>Lock-X(B)</td><td></td></tr></table>Which of the following is true?

a) Schedule is in dead-lock state.
b) Schedule is conflict serializable
c) Schedule is not conflict serializable
d) Both a & b</td><td>CO5</td><td>D</td></tr>
<tr><td><u>Q.No:4</u></td><td>MCQ</td><td>Let R = (A, B, C, D) be a relations schema with A, B, C, D are the candidate keys. The number of super keys formed are:
a) 4
b) 7
c) 15
d) 16</td><td>CO5</td><td>C</td></tr>
</table>

<table>
<tr><td></td><td>MCQ</td><td>Let R = (A, B, C, D, E) be a relation schema with {AB} is the only candidate keys.
The maximum number of super keys formed are:

(a) 3
(b) 4
(c) 5
(d) 8</td><td>CO4</td><td>D</td></tr>
<tr><td></td><td>MCQ</td><td>Consider the following set of functional dependencies, F = {A→B, A→C, A→D, B→C, B→E, C→E} defined on a relation schema R (A, B, C, D, E).
Which of the following is the set of redundant functional dependencies?

(a) A→B & A→C
(b) A→C & B→E
(c) A→B & B→C
(d) B→C & C→E</td><td>CO4</td><td>B</td></tr>
<tr><td></td><td>MCQ</td><td>Let R(A,B,C,D,E) be a relation schema with set of functional dependencies,
F = {AB→CDE, B→D, C→E}.
The relation R is decomposed into R1(B,D), R2(C,E) and R3(ABC). Which of the following is true for the above problem description?
I. R is in 1NF.
II. The decomposition is lossless join and dependency preserving decomposition.
III. The decomposition is lossless join and but not a dependency preserving decomposition.
IV. R1, R2, R3 are in BCNF

(a) Only I is true
(b) Both I & II are true
(c) Both I & III are true
(d) Only IV is true</td><td>CO4</td><td>B</td></tr>
<tr><td><u>Q.No:5</u></td><td>MCQ</td><td>Consider the following two tables:
LOAN
<table><tr><th>LoanNo</th><th>BranchName</th><th>Amount</th></tr><tr><td>L1001</td><td>KIIT</td><td>35000</td></tr><tr><td>L1002</td><td>IIT</td><td>42000</td></tr></table>
BORROWER</td><td>CO3</td><td>C</td></tr>
</table>

		<table><tr><th>CustmerNo</th><th>LoanNo</th></tr><tr><td>C1001</td><td>L1001</td></tr><tr><td>C1002</td><td>L1002</td></tr><tr><td>C1003</td><td>L1002</td></tr></table> SELECT * FROM LOAN, BORROWER; How many tuples will be resulted by the above query? (a) 4 (b) 5 (c) 6 (d) 7		
	MCQ	Consider the following Employee table: <table><tr><th>EmpName</th><th>BranchName</th><th>BranchCity</th><th>Salary</th></tr><tr><td>A</td><td>DU</td><td>DELHI</td><td>50000</td></tr><tr><td>B</td><td>DU</td><td>DELHI</td><td>55000</td></tr><tr><td>C</td><td>JNU</td><td>DELHI</td><td>60000</td></tr><tr><td>D</td><td>JU</td><td>KOLKOTA</td><td>65000</td></tr><tr><td>E</td><td>CU</td><td>KOLKOTA</td><td>70000</td></tr><tr><td>F</td><td>UU</td><td>BBSR</td><td>75000</td></tr></table>Find the distinct number of branches appearing in the Employee table. (a) select count (BranchName) from Employee; (b) select count (distinct (BranchName) from Employee; (c) select distinct count (BranchName) from Employee; (d) select count (*) from Employee;	CO3	B

	MCQ	Consider the following Employee table:
		<table><tr><th>EmpName</th><th>BranchName</th><th>BranchCity</th><th>Salary</th></tr><tr><td>A</td><td>DU</td><td>DELHI</td><td>50000</td></tr><tr><td>B</td><td>DU</td><td>DELHI</td><td>55000</td></tr><tr><td>C</td><td>JNU</td><td>DELHI</td><td>60000</td></tr><tr><td>D</td><td>JU</td><td>KOLKOTA</td><td>65000</td></tr><tr><td>E</td><td>CU</td><td>KOLKOTA</td><td>70000</td></tr><tr><td>F</td><td>UU</td><td>BBSR</td><td>75000</td></tr></table>Find the total salary of all employees at each branch of the bank?
		a) select BranchName, SUM(Salary) from Employee GROUP BY BranchCity;
		b) select BranchName, SUM(Salary) from Employee GROUP BY BranchName;
		c) select TOTAL(Salary) from Employee GROUP BY BranchName;
		d) select BranchName, SUM(Salary)
			CO3	B

		from Employee;		
	MCQ	Consider the following Employee table: <table><tr><th>EmpName</th><th>BranchName</th><th>BranchCity</th><th>Salary</th></tr><tr><td>A</td><td>DU</td><td>DELHI</td><td>50000</td></tr><tr><td>B</td><td>DU</td><td>DELHI</td><td>55000</td></tr><tr><td>C</td><td>JNU</td><td>DELHI</td><td>60000</td></tr><tr><td>D</td><td>JU</td><td>KOLKOTA</td><td>65000</td></tr><tr><td>E</td><td>CU</td><td>KOLKOTA</td><td>70000</td></tr><tr><td>F</td><td>UU</td><td>BBSR</td><td>75000</td></tr></table>Which of the following SQL statement will display all the employee records, whose salary is equal to the branch city wise average salary? a) select * from Employee where Salary in (select avg (Salary) from Employee group by Branchcity); b) select * from Employee where Salary = (select avg (Salary) from Employee group by Branchcity); c) select * from Employee where Salary is equal to (select avg (Salary) from Employee group by Branchcity); d) select avg (Salary) from Employee group by Branchcity;	CO3	A
Q.No:6	MCQ	I. The secondary index may be on a field which is a candidate key and has a unique value in every record, or a non-key with duplicate values. II. The secondary index is always based on a field which is a candidate key only. III. A secondary index provides a secondary means of accessing a file for which some primary access already exists Answer the following a) Statement I and II are true. b) Statement II and III are true. c) Statements I and III are true. d) All statements are true	CO6	C
	MCQ	Suppose we have an ordered file with 45,000 records stored on a disk with block size 2048 byte. File records are of fixed size with record length 120 bytes. What is the blocking factor? a) 17 b) 18 c) 21 d) 22	CO6	A
		In case of Log-Based Recovery of deferred database modification scheme, what		

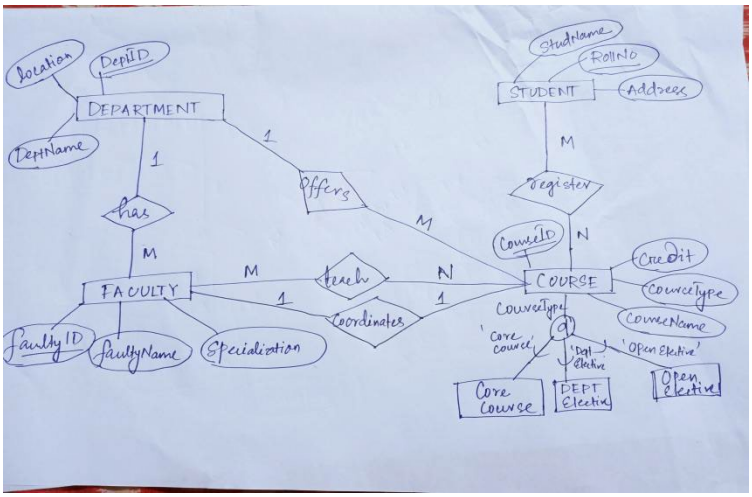
		action(s) to be taken after recovery of the system of the following schedule consist of two transactions T_0 and T_1? <T_0, start> <T_0, A, 1000, 900> <T_0, B, 2000, 2100> <T_0, commit> <T_1, start> <T_1, C, 750, 700> FAILURE a) Redo-T_0 and No action for T_1 b) Redo-T_0 and Redo-T_1 c) Redo-T_0 and Undo T_1 d) No action for T_0 and T_1	CO6	A
	MCQ	Which statement(s) is/are true with respect to two-phase locking protocol? I. Two transactions cannot have conflicting locks. II. No unlock operation can precede a lock operation in the same transaction. III. No data is/are affected until all locks are obtained and until the transaction is in its locked point. a) Only I b) Only II c) Both I & II d) All I, II, III	CO5	D
Q.No:7	MCQ	'Failures may leave database in an inconsistent state with partial updates carried out' is the case of a) Integrity problem b) Atomicity problem c) Security problem a) Data Redundancy & Inconsistency	CO1	B
	MCQ	Which statement(s) is/are not correct with respect to a database system requirement? I. High availability II. High response time III. High throughput a) Only I b) Only II c) Only III d) All are correct	CO5	B
	MCQ	Which of the following is used to provide faster access to data items stored in physical storage? a) Date files b) Indices		

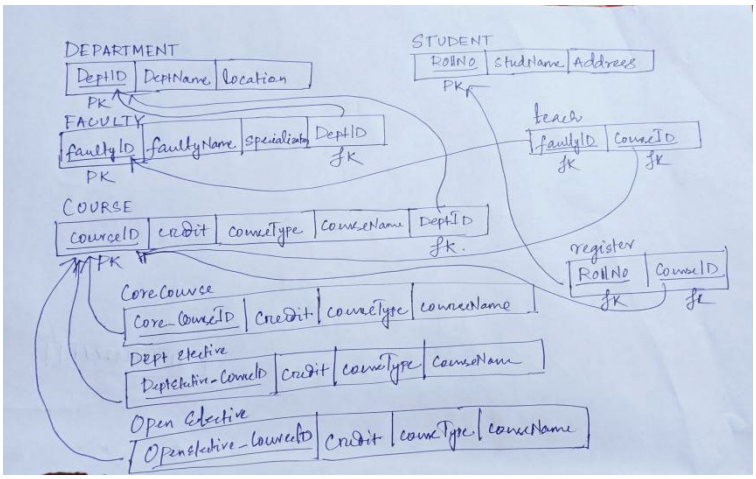
		c) Data dictionary d) Buffer manager	CO2	B
	MCQ	Consider a schema R(A, B, C, D) with set of functional dependencies, F={A→B, C→D}. Then the decomposition of R into R1 (A, B) and R2(C, D) is: a) dependency preserving and lossless join b) lossless join but not dependency preserving c) dependency preserving but not lossless join d) not dependency preserving and not lossless join	CO6	C

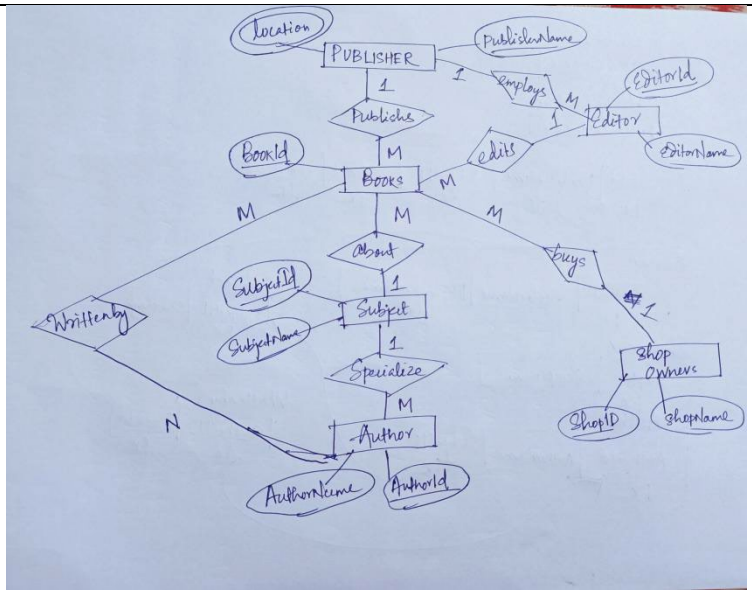
SECTION-B(Answer Any Three Questions. Each Question carries 12 Marks)

Time: 1 Hour and 30 Minutes **(3×12=36 Marks)**

Question No	Question	CO Mapping (Each question should be from the same CO(s))

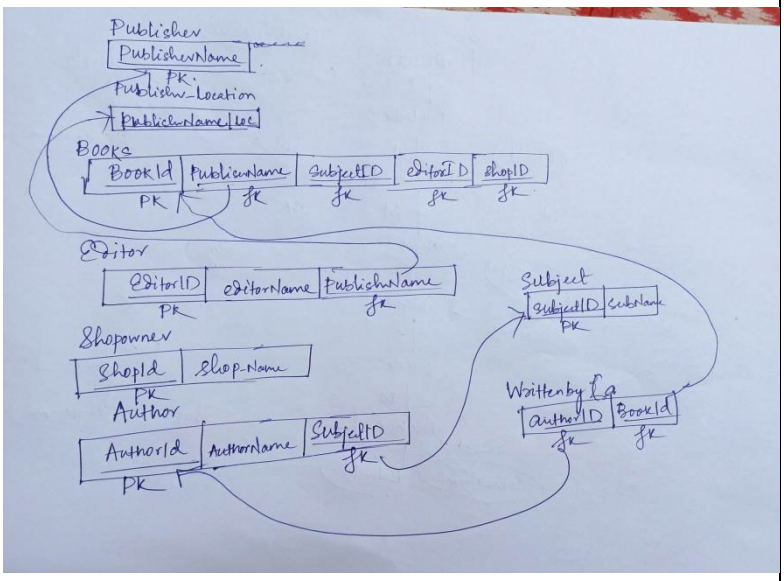
Q.No:8	**Consider the following problem description of a University:** A University has a number of Departments (identified by DeptID with other attributes as DeptName and Location). Each department offers a number of courses (identified by CourseID with other attributes as CourseName, Credit, CourseType) and a course cannot be offered by more than one department. Each department has a number of faculty members (identified by facultyID, with other attributes as FacultyName and Specialization) who works for that department but a faculty cannot work in more than one department. Students (identified by RollNo with other attributes as StudName, Address) register for courses and a course can also be registered by more than one student. Each course is either a Core Course or a Department Elective or an Open Elective course and this can be defined by an attribute called CourseType. A faculty can teach one or more than one course and a course can be taught by more than one faculty. Every course must have a course coordinator, who is in fact a faculty and a faculty can be the course coordinator of at most one course. a) Draw the ER/ Extended ER diagram of the above problem description. The given name of entities and attributes should be the same in the ER diagram. (Mention clearly all assumptions made by you in imposing constraints. **(8 marks)** **Solution**  	CO2

	b) Map the ER diagram into relations and specify the Primary keys and Foreign keys of each relation. The foreign keys must refer to the primary key using an arrow. **(4 marks)** **Solution**  	
Q. No. 8	**Consider the following problem description:** The publishing company (identified by PublisherName with other attributes as Location) produces books on various subjects (identified by subjectID with SubjectName as another attribute). Each publishing company is located in various locations. The books (identified by BookId) are written by authors (identified by AuthorID with AuthorName as another attribute) who specialize in one particular subject. The company employs editors (identified by EditorID with EditorName as another attribute) who, not necessarily being specialists in a particular area, each take sole responsibility for editing one or more book publications. Shop owners (identified by ShopID, with ShopName as another attribute) buy books from the publisher. Shop owners can buy many books but one book can be brought by one shop owner only. a) Draw the ER diagram of the above problem description. The given name of entities and attributes should be same in the ER diagram. (Mention clearly all assumptions made by you in imposing constraints. **(8 marks)** **Solution**	CO2



b) Map the ER diagram into relations and specify the Primary keys and Foreign keys of each relation. The foreign keys must refer to the primary key using an arrow. **(4 marks)**

Solution

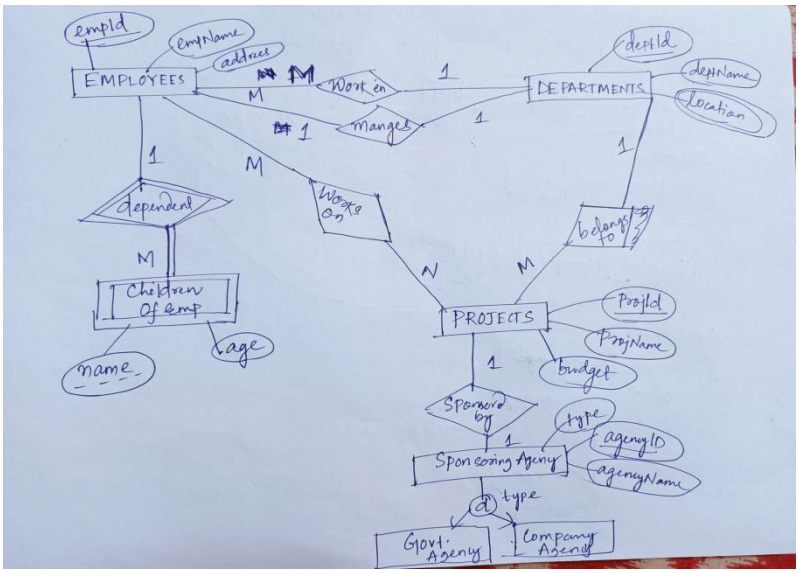


Q. No. 8	**Consider the following problem description:** An educational institute database needs to store information about employees (identified by empId, with empName, and address as attributes); departments (identified by deptId, with deptName and location as attributes); projects (identified by projId, with projName, budget as attributes) and children of employees (with name and age as attributes). A department can have many employees and an employee can work in only one department. Employees can works on different projects. A department can have many projects and a project can belongs to only one department. Each project is sponsored by one sponsoring agency (identified by agencyID with agencyName and type as attributes). The sponsoring agency can be either a Government Agency or a Company agency, whose information the Institute is interested to keep. The sponsoring	CO2

agencies are classified by the attribute 'Type'. Each department is managed by an employee. A department can be located in many places. A child must be identified uniquely by name when the parent (who is an employee; assume that only one parent works for the institute) is known. We are not interested in information about a child one the parent leaves the institute.

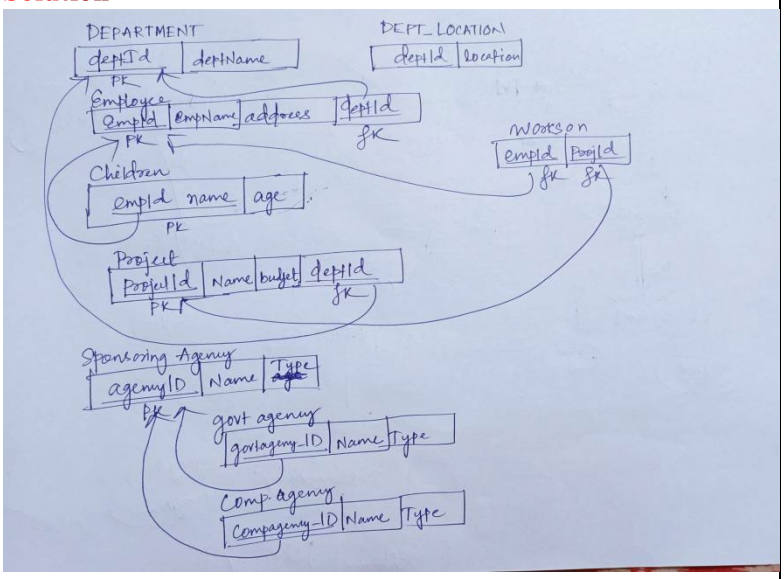
a) Draw the ER diagram of the above problem description. The given name of entities and attributes should be same in the ER diagram. (Mention clearly all assumptions made by you in imposing constraints. **(8 marks)**

Solution



b) Map the ER diagram into relations and specify the Primary keys and Foreign keys of each relation. The foreign keys must refer to the primary key using an arrow. **(4 marks)**

Solution



Q.No:9		
	i) Given relation schema, R (A, B, C, D) and set of functional dependencies, F = {B→A, AD→BC, C→ABD}. **[3+1+2+2=8 marks]** a) Compute the minimal cover for F. b) Find all candidate keys. c) Is the R in BCNF? If not, can we do BCNF decomposition? Justify. d) Is the R in 3NF?	CO4

Solution

i) Given relation schema, R (A, B, C, D) and set of functional dependencies,
F = {B→A, AD→BC, C→ABD}. [3+1+2+2=8 marks]

a. Compute the minimal cover for F. [3 MARKS]

Ans: F = {B→A, AD→BC, C→ABD}.

Step-1: Make RHS of each FD into a single attribute	*Step-2: Eliminate redundant attributes from LHS.*	*Step-3: Delete redundant FDs from F*
B→A AD→B AD→C C→A C→B C→D	Consider: {AD→B, AD→C} *Neither A nor D is redundant in AD→B and AD→C* B→A AD→B AD→C C→A C→B C→D	B→A AD→B AD→C C→A C→B C→D The canonical cover is {B→A, AD→C, C→BD}

b. Find all candidate keys. [1 MARK]
C, AD, BD
(As $C^+ = ABCD$, $AD^+ = ADBC$, $BD^+ = ABCD$)

c. Is the R is in BCNF? If not, can we do BCNF decomposition? Justify. [2 MARKS]
No, No decomposition is possible as AD→BC cannot be preserved..

d. Is R is in 3NF? [2 MARKS]
Yes, It is in 3NF as all fds are satisfied

ii) State and prove Armstrong's Union and Decomposition rules using Armstrong's 3 axioms.

[4marks]

Solution

1. Union: [2 MARKS]

- ***If $X \rightarrow Y$, and $X \rightarrow Z$, then $X \rightarrow YZ$***

Proof: $X \rightarrow Y \Rightarrow XX \rightarrow XY$ *(augmenting X on both sides)*

$\Rightarrow X \rightarrow XY$ --- *(1)*

$X \rightarrow Z \Rightarrow YX \rightarrow YZ$ *(Aug. Rule)* --- *(2)*

Eq. (1) & (2) implies: $X \rightarrow YZ$. *Proved*

2. Decomposition: [2 MARKS]

- *If $X \rightarrow YZ$ then $X \rightarrow Y$, and $X \rightarrow Z$.*

Proof: $Y \subseteq YZ \Rightarrow YZ \rightarrow Y$ (Reflexivity Rule) ---(1)

$X \rightarrow YZ$ (Given) & Eq.(1) $\Rightarrow X \rightarrow Y$ Proved.

Similarly,

$Z \subseteq YZ \Rightarrow YZ \rightarrow Z$ (Reflexivity Rule) ---(2)

$X \rightarrow YZ$ (Given) & Eq.(2) $\Rightarrow X \rightarrow Z$ Proved.

Q.No:9

i) Let R (A,B,C,D,E,F) be relation schema with a set of functional dependencies, G = {A→B, C→DF, AC→E}. **[2+2+4=8 marks]**

a)Find the candidate key(s)

b) Do the BCNF decomposition.

c)Check whether the decomposition is lossless join and dependency preserving decomposition or not?

CO4

Solution

a) Find the candidate key(s) **[2 marks]**

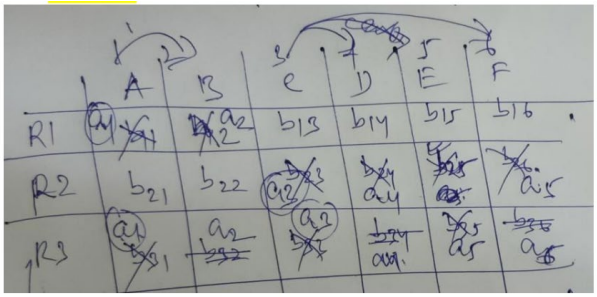
{AC} is the candidate key

b) Do the BCNF decomposition. **[2 marks]**

R(A,B,C,D,E,F) *G = {A→B, C→DF, AC→E}*

R1(A,B) *R2(C,D,F)* *R3(A,C,E)* *All are in BCNF*

A→B C→DF



Since all third row contains all a_i's. So it is lossless join decomposition
It is also a dependency preserving decomposition..

ii) Let R (A, B, C, D, E, G, H be a relation schema with set of functional dependencies, F = {A→B, B→A, C→D, E→GD, C→A, H→ CAB, EG→ CA}. Using all the required steps, find the canonical cover of F. **[4 marks]**

Solution

F={A→B, B→A, C→D, E→GD, C→A, H→ CAB, EG→ CA}.

Step-1: Make RHS of each FD into a single attribute	*Step-2: Eliminate redundant attributes from LHS.*	*Step-3: Delete redundant FDs from F*
A→B B→A C→D E→G E→D C→A	Consider: {EG→ C, EG→ A} E^+ = EGDAB G^+ = G E^+ = EGDCA In EG→A, G is redundant??	A→B B→A C→D E→G E→D C→A

H→ C H→ A H→ B EG→ C EG→ A	So, replace EG→A with E→A A→B B→A C→D E→G E→D C→A H→ C H→ A H→ B EG→ C E→ A	H→ C H→ A H→ B EG→ C E→ A The canonical cover is {A→B, B→A, C→D, E→G, C→A, H→ C EG→ C

Q.No:9	i) Find the minimal cover of the given set of functional dependencies, F = {AB→E, C→D, F→GH, FG→GH, B→FG} using step by step process. **[4 marks]** **Solution** *i) The Minimal Cover:* F = {AB→E, C→D, F→GH, FG→GH, B→FG} [4 MARKS]	CO4

Step-1: Make RHS of each FD into a single attribute	*Step-2: Eliminate redundant attributes from LHS.*	*Step-3: Delete redundant FDs from F*
AB→E C→D F→G F→H FG→G FG→H B→F B→G	Consider: {AB→E, FG→G, FG→H} The G is redundant in FG→G (F→G) The G is redundant in FG→H (F→H) AB→E C→D F→G F→H F→G F→H B→F B→G	AB→E C→D F→G F→H F→G F→H B→F B→G The canonical cover is {AB→E, C→D, F→G, B→F}

ii) State and prove the pseudo transitivity rule using Armstrong's axioms. **[2 marks]**

Solution

3. **Pseudo-transitivity:**

- *If $X \rightarrow Y$ and $WY \rightarrow Z$ then $WX \rightarrow Z$*

Proof:

$X \rightarrow Y$ (Given) $\Rightarrow WX \rightarrow WY$ (Aug Rule) ---(1)

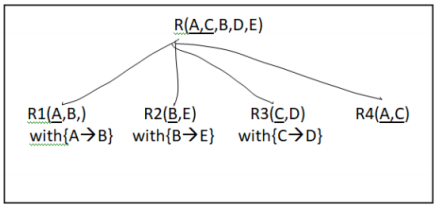
Eq.(1) and $WY \rightarrow Z$ (Given) $\Rightarrow WX \rightarrow Z$. *Proved.*

iii) Given a relation R (A, B, C, D, E) and set of functional dependencies, F = { A→B, B→E, C→D}
[1+5= 6 Marks]

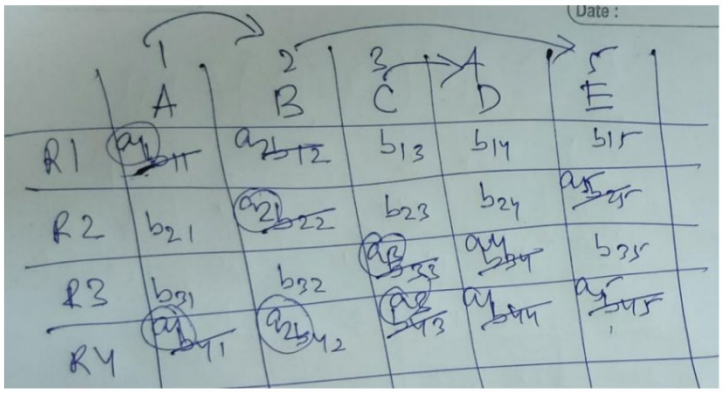
a) Find the candidate key.
b) Is R in 3NF? If not, then convert it into 3NF and check for the lossless join property of the decomposition.

Solution

a) The Candidate key: {AC} [1 Marks]
b) R is NOT in 3NF [1+2+2 =5 Marks]
R is decomposed into R1(AB), R2(BE), R3(CD), R4(AC) as follows:



Checking for the lossless join property



Since row R4 having all ai's, so it is a loss less join decomposition

Q.No:10

Consider the following concurrent schedule consists of transactions T1 and T2:

T1	*T2*
Read(X)	
X=X-100	
	Read(X)
	X= X+150
Write (X)	
Read(Y)	
	Write(X)
Y=Y+100	
Write(Y)	

The initial values of X and Y are 1000 and 2000 respectively.
[2+2+4+4=12 marks]

a) Find the final values of X and Y. (mention outputs in a step by step manner, using above diagram)

Solution

T1	T2	Primary Memory	Database
Read(X)		X=1000 by T1	X=1000 Y=2000
X=X – 100		X=900 by T1	
	Read (X)	X=1000 by T2	
	X= X + 150	X=1150 by T2	
Write(X)			X=900 by T1
Read(Y)		Y=2000	
	Write(X)		X=1150 by T2
Y=Y+100		Y=2100	
Write(Y)			Y=2100 by T1

The FINAL VLUE of X=1150 & Y=2100

b) If the transactions are executed in serial manner <T1,T2>, then what would be the values of X and Y at the end of the given serial execution? (mention the outputs in a step by step manner, by drawing the serial schedule <T1,T2> diagram) [2 marks]

Solution

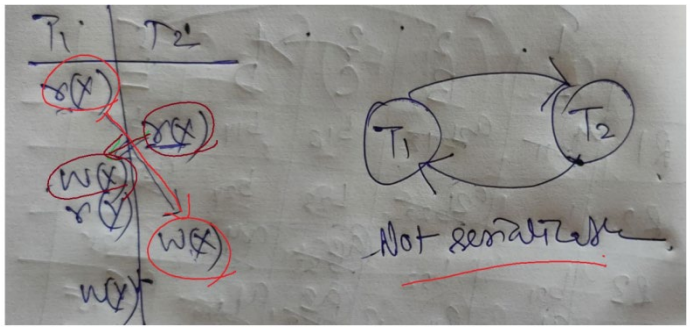
CO5

The Schedule <T1, T2>

T1	T2	Primary Memory	Database
Read(X)		X=1000	X=1000 Y=2000
X=X – 100		X=900	
Write(X)			X=900
Read(Y)		Y=2000	
Y=Y+100		Y=2100	
Write(Y)			Y=2100
	Read (X)	X=900	
	X= X + 150	X=1050	
	Write (X)		X= 1050

The FINAL VLUE of X=1050 & Y=2100

c) Is the given concurrent schedule is conflict serializable? Test using precedence graph.



d) What would be the scenario, if data-locks (shared and exclusive) are given to data items in the given concurrent schedule? Answer by drawing the updated schedule with all locks.

Solution

Scenario-1 (Lock Shared/Exclusive to be given) [4 marks]

T1	T2	CC Manager
Lock-X(X) Read(X)		GRANT-X (X,T1) Lock Granted
X=X – 100		
	Lock-X(X) Read (X)	GRANT-X (X,T2) WAIT??? DEADLOCK Situation
	X= X + 150	
Write(X)		
Lock-X(Y) Read(Y)		GRANT-X (Y,T1)
	Write(X)	
Y=Y+100		
Write(Y)		

Scenario-2 (Lock Conversion)

T1	T2	CC Manager	ACTION
Lock-X(X) Read(X)		GRANT-S (X,T1)	Granted
X=X - 100			
	Lock-X(X) Read (X)	GRANT-S (X,T2)	Granted
	X= X + 150		

Write(X)		Upgrade (X)	Wait
Lock-X(Y) Read(Y)			
	Write(X)		
Y=Y+100			
Write(Y)			

Q.No:10	i) Consider the two transactions (T1 & T2) on a bank account, 'B', with initial balance Rs. 15000/-, that transfer Rs. 3000/- to a mortgage, 'M', payment (by T1) and then apply 5% interest to account 'B' (by T2). Initially M is empty. **[4+2=6 marks]** a)Create the log sequence of the serial schedule <T1, T2>	CO5

Solution

a) Log Sequence of <T1,T2> **[4 Mraks]**

T1	T2	Primary Memory	DATABASE	Log
			B=15000 M=0	
Read(B)		B=15000		*<T1, START>*
B=B-3000		B=12000		
Write (B)			B=12000	*<T1, B, old= 15000, new=12000>*
Read(M)		M=0		
M=M+3000		M=3000		
Write(M)			M=3000	*<T1, M, old= 0, new=3000>*
COMMIT				*<T1, COMMIT>*
	Read(B)	b=12000		*<T2, START>*
	*B=B*1.05*	B=12600		
	Write(M)		B=12600	*<T2, B, old=12000, new=12600>*
	COMMIT			*<T2, COMMIT>*

b) Find the actions to be taken when the database system crashes just before the COMMIT operation (and after the last statement) of T2 with the Deferred Updates.

Solution

Deferred UPDATE

Log
<T1, START>
<T1, B, old= 15000, new=12000>
<T1, M, old= 0, new=3000>
<T1, COMMIT>
<T2, START>
<T2, B, old=12000, new=12600>
<T2, COMMIT>

Failure ⇨

At failure point: ACTION: "Redo(T1), No Action for T2"

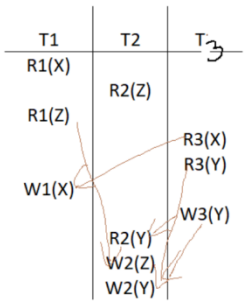
ii) Consider three transactions T1, T2, T3 and schedule S1as given below.
T1: r1(x), r1(z), w1(x)
T2: r2(z), r2(y), w2(z), w2(y)
T3: r3(x), r3(y), w3(y)
S1: r1(x), r2(z), r1(z), r3(x), r3(y), w1(x), w3(y), r2(y), w2(z), w2(y).

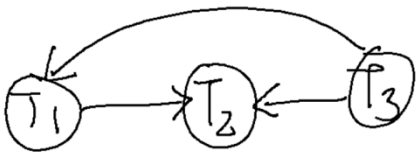
[4+2=6 marks]

a)Check the conflict serializability of S1 by drawing the precedence graph.

Solution

a) Schedule S1: [4 MARKS]





NO Cycle implies SERIALIZABLE

b) If the schedule is conflict serializable, then find the equivalent serial schedule.

Solution

b) Equivalent Serial Schedule [2 MARKS]

Q.No:10	i) Is every view serializable schedule is conflict serializable? Explain with a suitable example. **[4 marks]** **Solution** *i) The answer is NO... "Every conflict serializable schedule is also view serializable, but the reverse is not always true".* *Any suitable example can be given with blind write:* T1 / T2 / T3 Read(Q) / Write(Q) / Write(Q) ii) Check the view serilaizability of the given schedule {R1(X); W1(X); R3(X); R2(X); W3(X)} using label precedence graph. **[4 marks]** **Solution** *ii) View Serializability of the given schedule (the digarm used by Adhyasa also considered)* T1 / T2 / T3 R1(X) W1(X) / R2(X) / R3(X) W3(X) Tb: W(X) T1: R1(X) T1: W1(X) T3: R3(X) T2: R2(X) T3: W3(X) Tf: R(X) NO Cycle implies View Serializable iii) Can you apply two phase locking protocol on the given schedule {R1(X); W1(X); R3(X); R2(X); W3(X)}? Justify your answer with depicting locks in the diagram of given concurrent schedule. **[4 marks]** **Solution**	CO5

iii) Two Phase Locking can be applied with Lock conversion as follow:

T1	T2	T3	CC Manager
Lock-Exclusive (X)			GRANT-Exclusive (X,T1)
R1(X) W1(X) Unlock (X)			
		Lock-S (X)	GRANT-S (X, T3)
		R3(X)	
	Lock-S (X)		GRANT-S (X, T2)
	R2(X) Unlock(X)		
		UPGRADE (X)	Upgraded to Exclusive Lock (X, T3)
		W3(X) Unlock(X)	

Q.No:11

CO4 & CO6

i. Consider the following tables of a relational database system
HOTEL (hotelNo, hotelName, city)
ROOM (roomNo, hotelNo, type, price)
BOOKING (hotelNo, guestNo, dateFrom, dateTo, roomNo)
GUEST (guestNo, guestName, guestAddress)
Answer the following queries using SQL:

[4 x 2 = 8 marks]

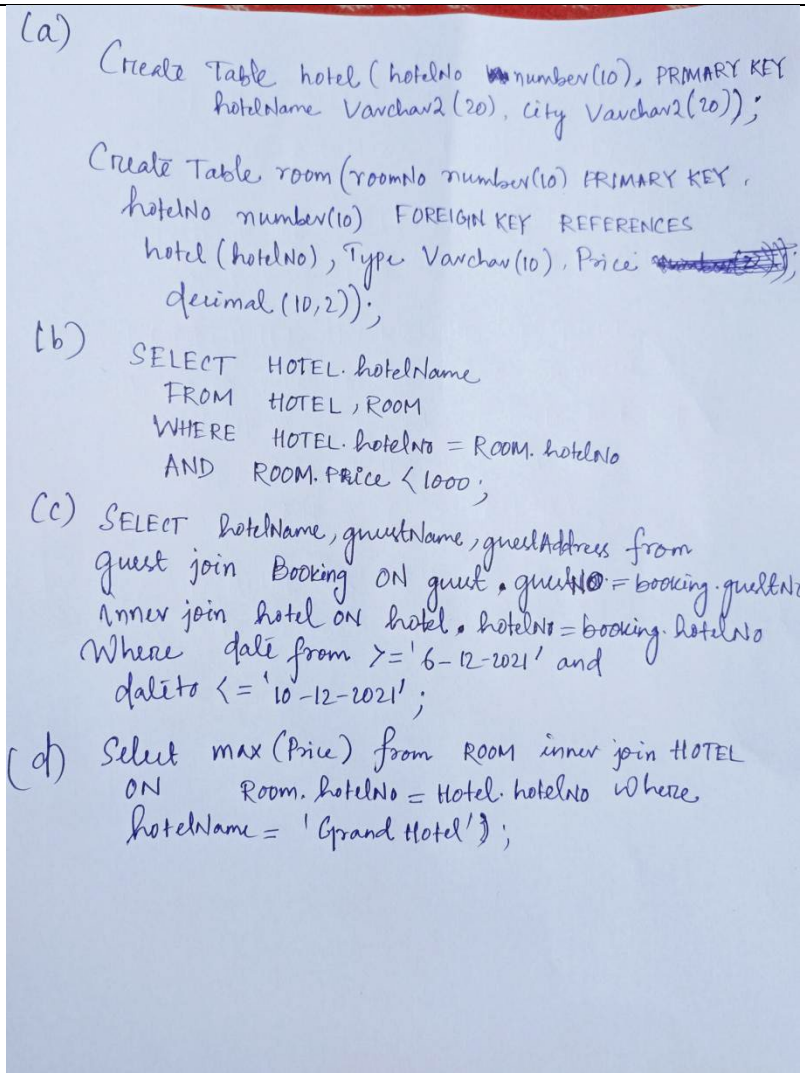
a) Create HOTEL and ROOM tables with primary key and foreign key constraints.

b) List name of all hotels whose room price is below Rs. 1000/- per night.

c) Display the hotel name, guest name and guest address of all guests who have booked rooms from 6-12-2021 to 10-12-2021.

d) Display the highest room price of the hotel named as 'Grand Hotel'.

Solution



ii. Supported with suitable example (of your own) and schematic diagram, explain the primary index and secondary index. **(4 marks)**

Solution

Primary Index in DBMS

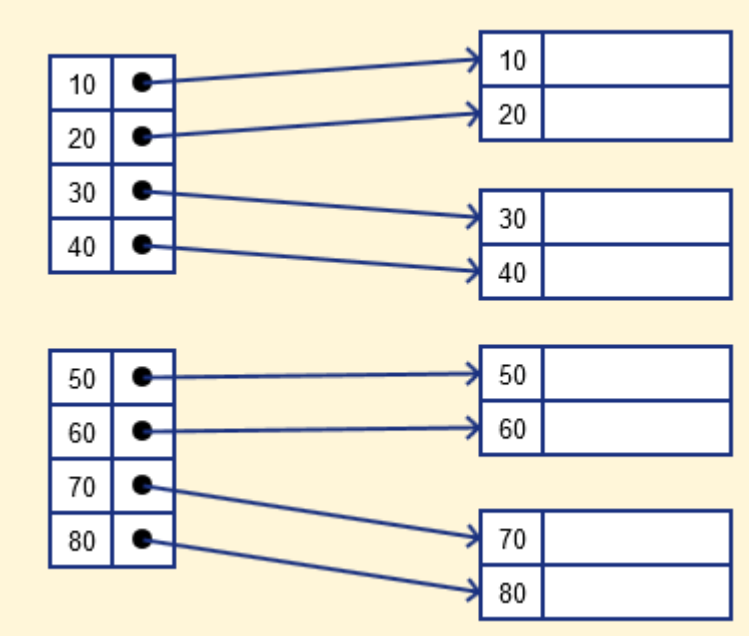
Primary Index is an ordered file which is fixed length size with two fields. The first field is the same a primary key and second, filed is pointed to that specific data block. In the primary Index, there is always one to one relationship between the entries in the index table.

The primary Indexing in DBMS is also further divided into two types.

- Dense Index
- Sparse Index

Dense Index

In a dense index, a record is created for every search key valued in the database. This helps you to search faster but needs more space to store index records. In this Indexing, method records contain search key value and points to the real record on the disk.

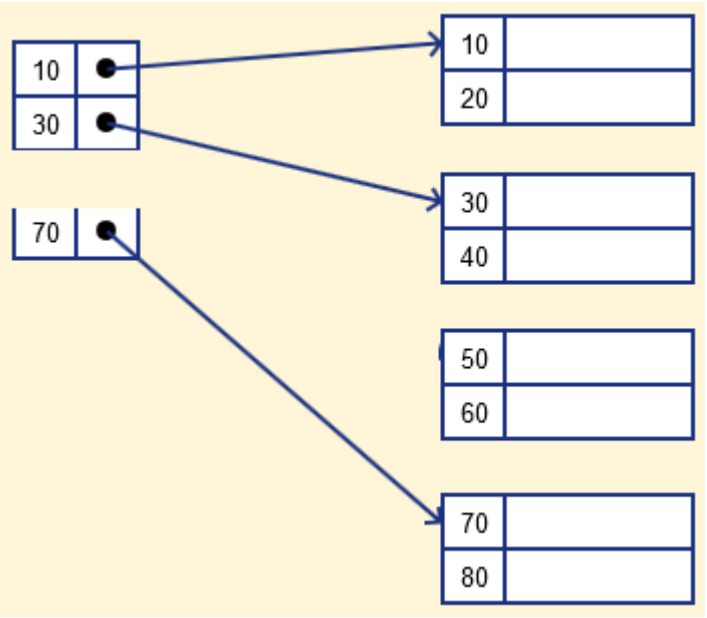


Sparse Index

It is an index record that appears for only some of the values in the file. Sparse Index helps you to resolve the issues of dense Indexing in DBMS. In this method of indexing technique, a range of index columns stores the same data block address, and when data needs to be retrieved, the block address will be fetched.

However, sparse Index stores index records for only some search-key values. It needs less space, less maintenance overhead for insertion, and deletions but It is slower compared to the dense Index for locating records.

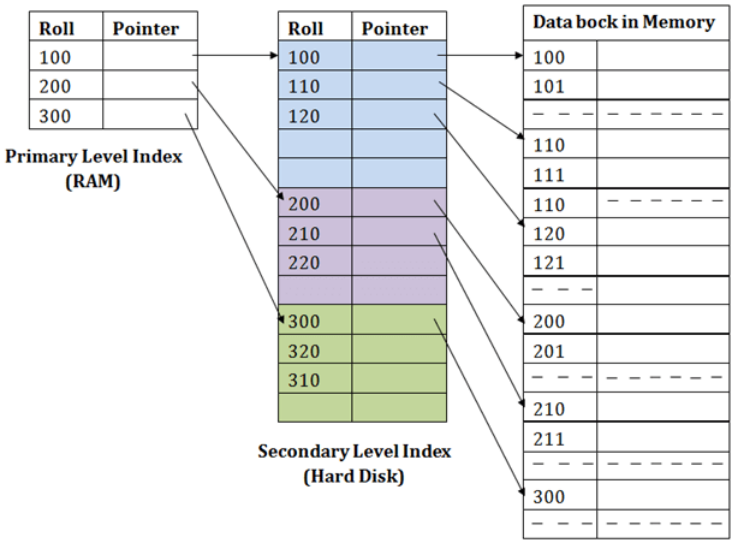
Below is an database index Example of Sparse Index



Secondary Index

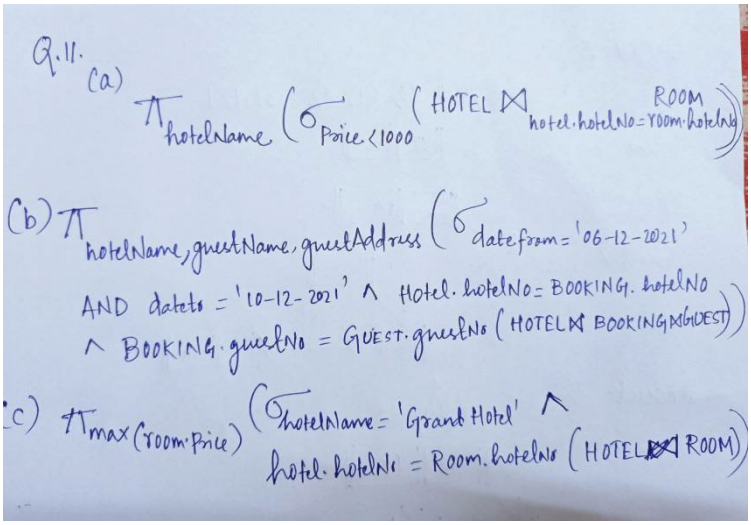
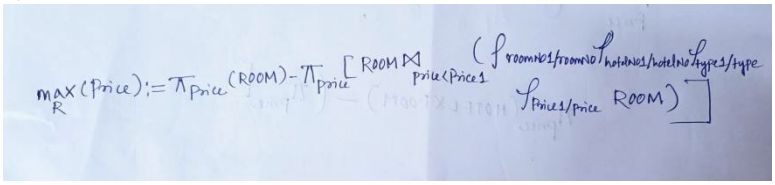
In the sparse indexing, as the size of the table grows, the size of mapping also grows. These mappings are usually kept in the primary memory so that address fetch should be faster. Then the secondary memory searches the actual data based on the address got from mapping. If the mapping size grows then fetching the address itself becomes slower. In this case, the sparse index will not be efficient. To overcome this problem, secondary indexing is introduced.

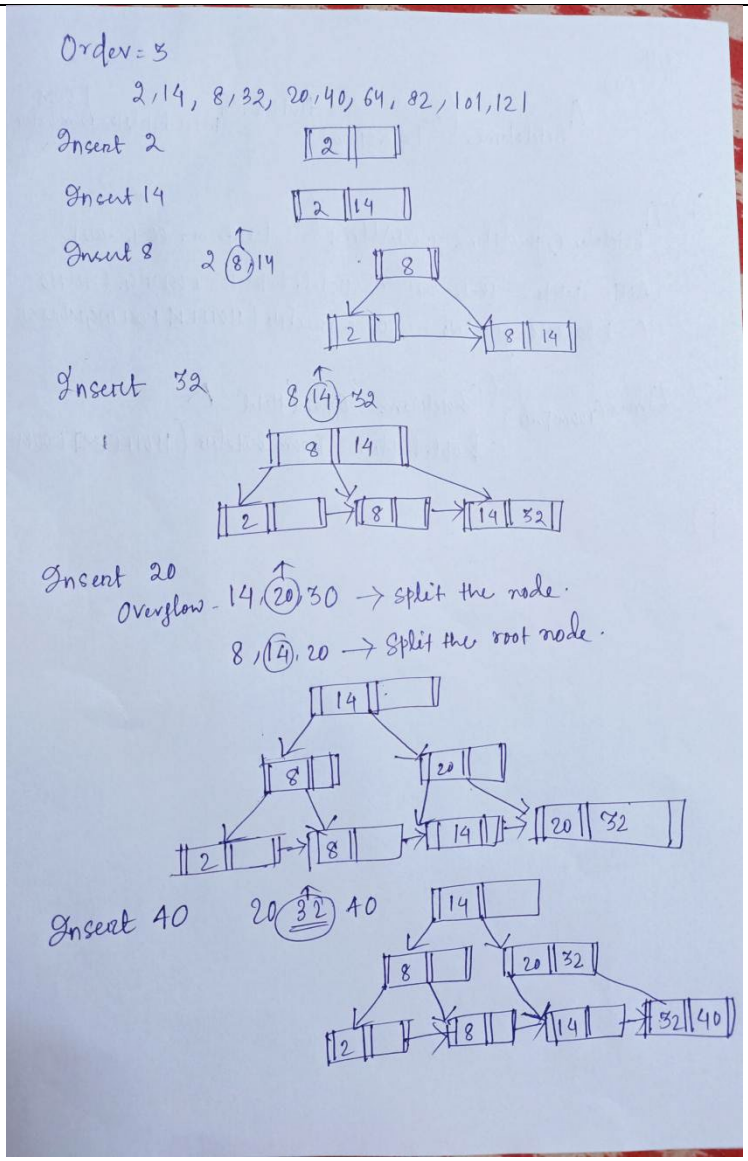
In secondary indexing, to reduce the size of mapping, another level of indexing is introduced. In this method, the huge range for the columns is selected initially so that the mapping size of the first level becomes small. Then each range is further divided into smaller ranges. The mapping of the first level is stored in the primary memory, so that address fetch is faster. The mapping of the second level and actual data are stored in the secondary memory (hard disk).

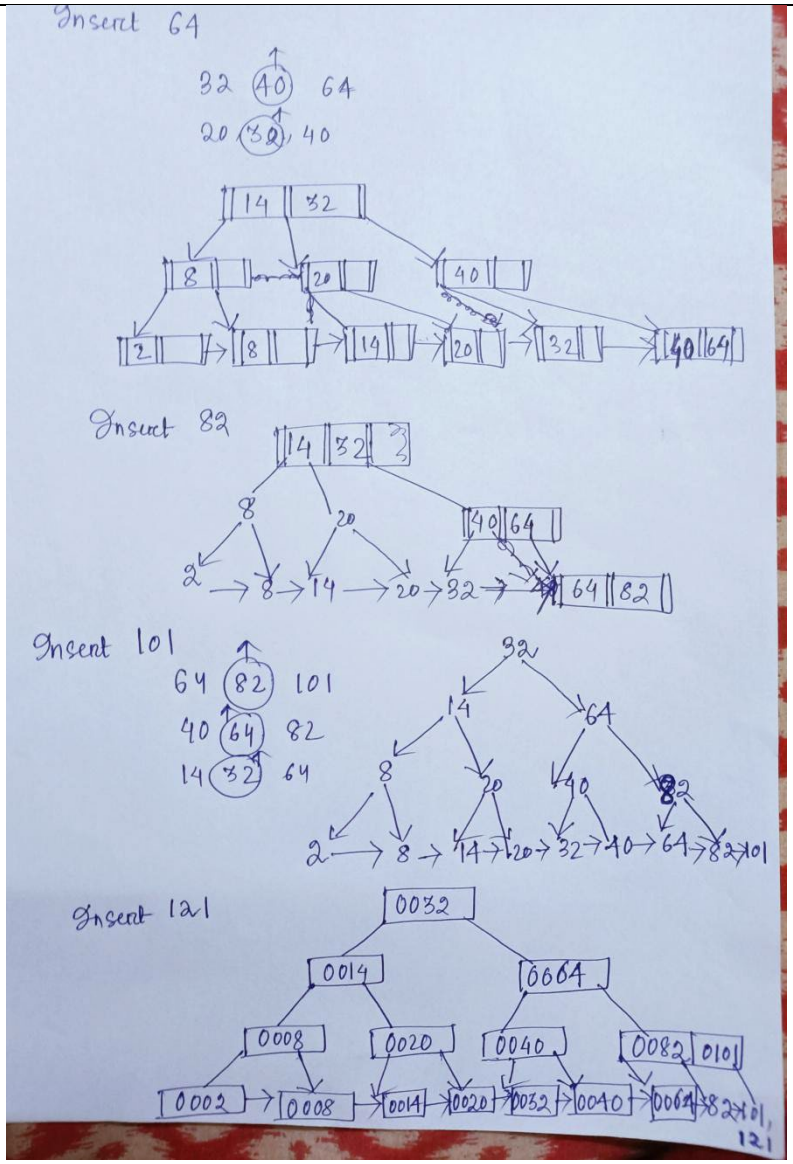


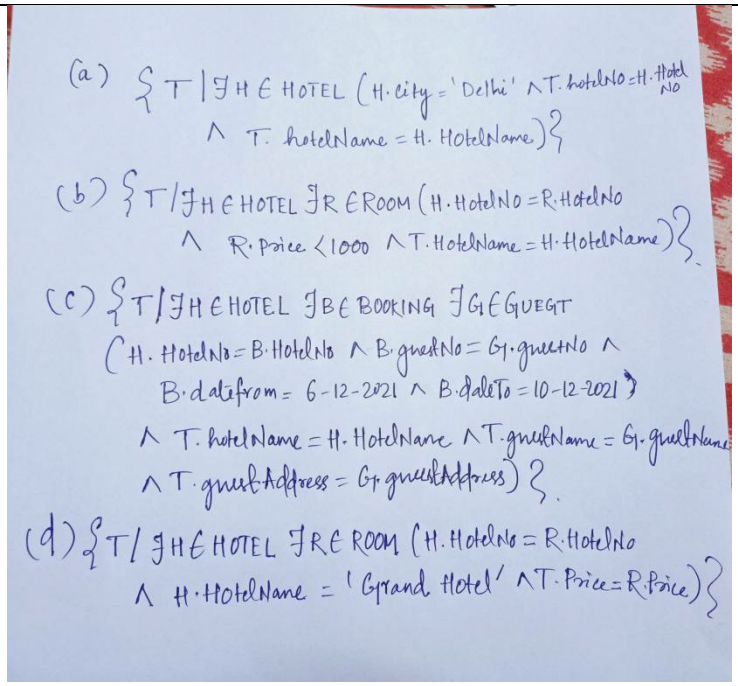
For example:

- ✓ If you want to find the record of roll 111 in the diagram, then it will search the highest entry which is smaller than or equal to 111 in the first level index. It will get 100 at this level.
- ✓ Then in the second index level, again it does max (111) <= 111 and gets 110. Now using the address 110, it goes to the data block and starts searching each record till it gets 111.
- ✓ This is how a search is performed in this method. Inserting, updating or deleting is also done in the same manner.

Q.No:11	i. Consider the following tables of a relational database system HOTEL (<u>hotelNo</u>, hotelName, city) ROOM (<u>roomNo, hotelNo</u>, type, price) BOOKING (<u>hotelNo, guestNo, dateFrom</u>, dateTo, roomNo) GUEST (<u>guestNo</u>, guestName, guestAddress) Answer the following queries using Relational Algebra: **[4 x 2 = 8 marks]** a) List name of all hotels whose room price is below Rs. 1000/- per night. b) Display the hotel name, guest name and guest address of all guests who have booked rooms from 6-12-2021 to 10-12-2021. c) Display the highest room price of the hotel named as 'Grand Hotel'. d) Display the highest room price of all hotels without using max() function. **Solution**  d)  iii. Construct a B+ tree of order 3, for (2, 14, 8, 32, 20, 40, 64, 82, 101, 121). Mention all steps for every insertion during the creation of the tree. **(4 marks)** **Solution**	CO4 & CO6


Order = 5
2, 14, 8, 32, 20, 40, 64, 82, 101, 121
Insert 2
Insert 14
Insert 8
2 (8) 14
Insert 32
8 (14) 32
Insert 20
Overflow - 14, (20) 30 → split the node.
8, (14), 20 → Split the root node.
Insert 40
20 (32) 40

		
Q.No:11	i. Consider the following tables of a relational database system HOTEL (hotelNo, hotelName, city) ROOM (roomNo, hotelNo, type, price) BOOKING (hotelNo, guestNo, dateFrom, dateTo, roomNo) GUEST (guestNo, guestName, guestAddress) Answer the following queries using tuple relational calculus: **[4 x 2 = 8 marks]** a) Display all hotels which are located in 'Delhi'. b) List name of all hotels whose room price is below Rs. 1000/- per night. c) Display the hotel name, guest name and guest address of all guests who have booked rooms from 6-12-2021 to 10-12-2021. d) Display the room price of the hotel named as 'Grand Hotel'. Solution	CO4 & CO6

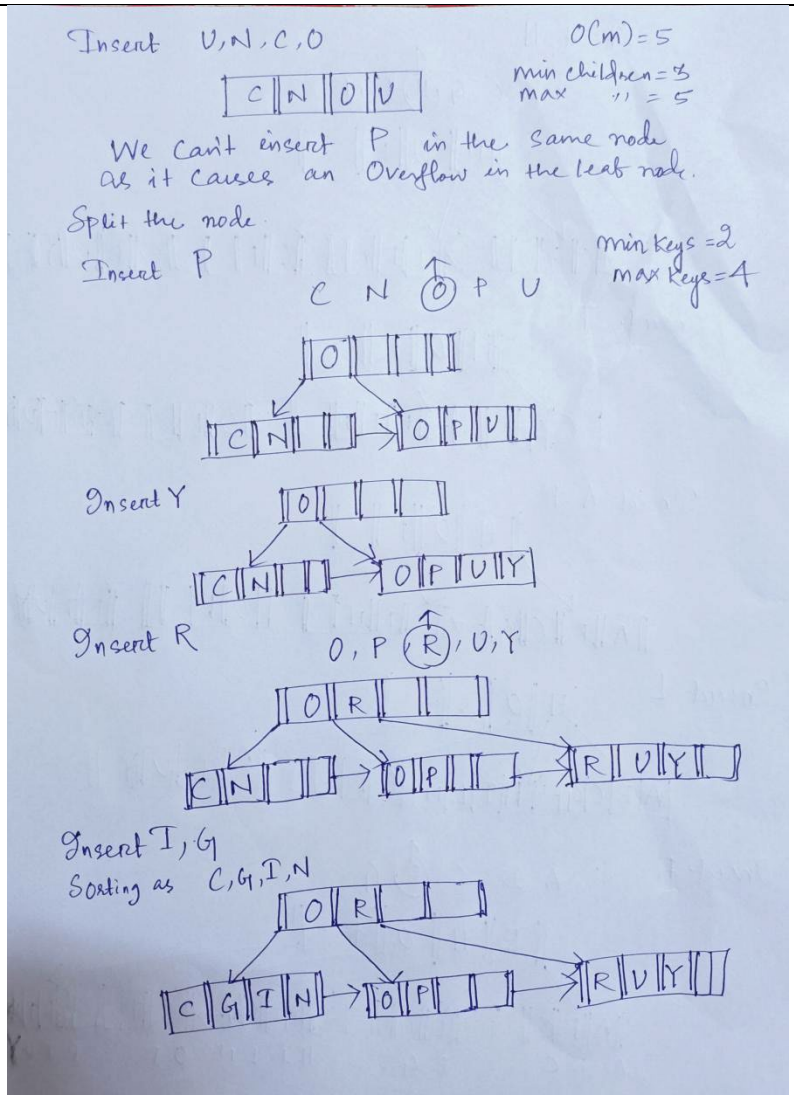


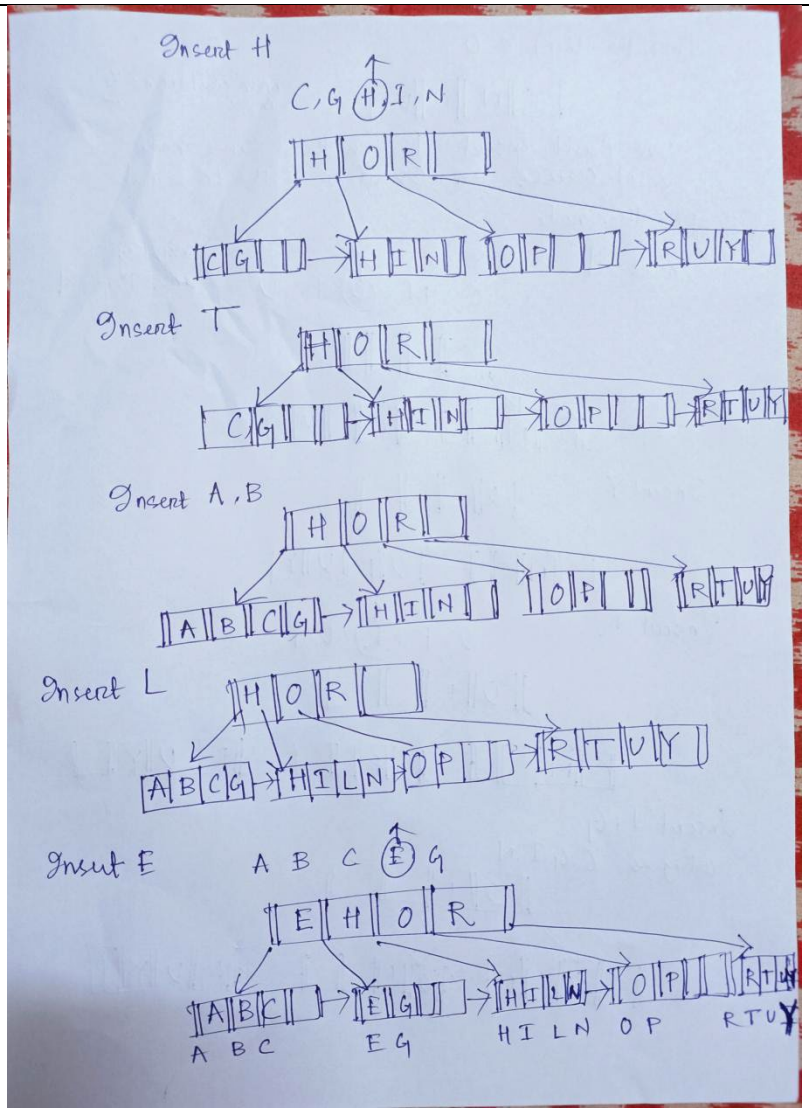
ii) Construct a B+ tree of order 5, for letters (U, N, C, O, P, Y, R, I, G, H, T, A, B, L, E) Mention all steps for every insertion during the creation of the tree. **(4 marks)**

Solution

The order is 5 so at maximum in a node so there can be only 4 search key values.

As insertion happens on a leaf node only in a B+ tree so insert search key letters in increasing order in the node. Below is the illustration of the same:


Insert U, N, C, O
O(m) = 5
min children = 3
max " = 5
C N O U
We can't insert P in the same node as it causes an Overflow in the leaf node.
Split the node
Insert P
min keys = 2
max keys = 4
C N O P U
O
C N O P U
Insert Y
O
C N O P U Y
Insert R
O, P, R, U, Y
O R
C N O P R U Y
Insert I, G
Sorting as C, G, I, N
O R
C G I N O P R U Y

CO #	Detail
CO1	Describe the fundamental elements of relational database management systems and understand the database systems and its applications.
CO2	Conceptualize and depict a database system using ER diagram.
CO3	Construct queries using relational algebra, relational calculus and SQL.
CO4	Understand the functional dependencies and design the database using normalization.
CO5	Understand the needs of Transaction processing and learn techniques for controlling the consequences of concurrent data access.
CO6	Understand basic database storage structures and access techniques: file organizations, indexing methods including B-tree, and hashing.