

Name: Kamallesh S

Regn. No: 2127200801040

Subject: Database System

Section: A Branch: Information Technology

Assignment - 2

- 1) Let $r(z)$ & $w(z)$ denote read & write operations respectively on a data item z by a transaction T . Consider a schedule:-

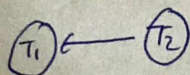
S1: $r_1(x)$ $r_1(y)$ $r_2(x)$ $r_2(y)$ $w_2(y)$ $w_1(x)$

S2: $r_1(x)$ $r_2(x)$ $r_2(y)$ $w_2(y)$ $r_1(y)$ $w_1(x)$

Which schedule is conflict serializable? how?

Sol: S2 is conflict serializable. This is because the transaction series of S2 looks like this

T_1	T_2
$r_1(x)$	$r_2(x)$
	$r_2(y)$
	$w_2(y)$
$r_1(y)$	
$w_1(x)$	



- 2) Consider the relation $R(P, Q, S, T, X, Y, Z, W)$ with following functional dependencies $PQ \rightarrow X$, $P \rightarrow Y$, $Q \rightarrow Y$, $Y \rightarrow Z$. Consider the decomposition of Relation R into the relations $D_1 = \{P, Q, S, T\}$, $\{P, T, X\}$, $\{Q, Y\}$, $\{Y, Z, W\}$
 $D_2 = \{P, Q, S\}$, $\{T, X\}$, $\{Q, Y\}$, $\{Y, Z, W\}$. Identify the type of decomposition (lossy / lossless). Justify.

Sol: D_1 is a lossless decomposition, but D_2 is a lossy decomposition.

$D_1: R = \{P, Q, S, T\}, \{P, T, X\}, \{Q, Y\}, \{Y, Z, W\}$

Take $R_1(P, Q, S, T)$ & $R_2(P, T, X)$, common attributes are P & $T \rightarrow TX$

$\Rightarrow R_1(P, Q, S, T, X)$ $R_2(Q, Y)$

$Q \rightarrow Y \Rightarrow R_1(P, Q, S, T, X, Y)$

relation becomes $R_1(P, Q, S, T, X, Y)$ $R_2(Y, Z, W)$

common attribute is Y & its key to R_2 .

Hence, all attributes got combined into one relation & hence this decomposition is lossless.

$D_2: R = [(P, Q, S), (T, X), (Q, Y), (Y, Z, W)]$

In relation (T, X) , its attribute is not common to any other relation.
Even if we combine

Since $R_1(P, Q, S, Y, Z, W)$ $R_2(T, X)$, still no common attributes

\therefore this decomposition is lossy.

- 3) Suppose following functional dependency hold on a relation V with attributes P, Q, R, S, T .

$P \rightarrow QR$

$RS \rightarrow T$

What FD can be inferred from above? explain.

Sol:

$PS \rightarrow T$

~~that~~ $P \rightarrow R$

$PS \rightarrow Q$

} These are the inferred FDs.

From $P \rightarrow QR$, we can derive $P \rightarrow Q$ & $P \rightarrow R$

$P \rightarrow R \Rightarrow PS \rightarrow T$

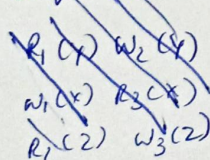
$P \rightarrow Q \Rightarrow PS \rightarrow Q$

- 4) Consider the schedule of 3 transactions
 $R_2(Y)$ $R_1(X)$ $R_3(Z)$ $R_1(Y)$ $W_1(X)$ $R_2(Z)$ $W_2(Y)$ $R_3(X)$ $W_3(Z)$. Is this schedule a conflict serializable.

Sol:

For conflict serializability of a schedule (which gives same effect as a serial schedule) we should check for conflict operations, which are read-write, write-read & write-write between each pair of transactions, & based on their conflicts we make a precedence graph, if graph contains a cycle, it's not a conflict serializable schedule.

Conflicts:-



$R_1(X)$ $R_2(Y)$ $R_3(Z)$ $R_1(Y)$ $W_1(X)$ $R_2(Z)$ $W_2(Y)$ $R_3(X)$ $W_3(Z)$

$R_1(X)$ $R_1(Y)$ $R_3(Z)$ $R_2(Y)$ $W_1(X)$ $R_2(Z)$ $W_2(Y)$ $R_3(X)$ $W_3(Z)$

$R_1(X)$ $R_1(Y)$ $W_1(X)$ $R_2(Y)$ $R_3(Z)$ $R_2(Z)$ $W_2(Y)$ $R_3(X)$ $W_3(Z)$

$R_1(X)$ $R_1(Y)$ $W_1(X)$ $R_2(Y)$ $R_2(Z)$ $W_2(Y)$ $R_3(Z)$ $R_3(X)$ $W_3(Z)$

\therefore it is conflict serializable of $T_1 \rightarrow T_2 \rightarrow T_3$.

Justify the following statements.

Strict 2PL protocol generates conflict serializable schedules that are also recoverable.
Timestamp ordering concurrency protocol with Thomas write rule can generate view serializable schedules that are not conflict serializable.

Sol: Following Strict 2-PL ensures that our schedule is: serializable, recoverable & cascadeless.

Thomas write rule for concurrency control does not enforce conflict serializability. In 2PL, transactions do not release exclusive locks until the transaction has committed or aborted, i.e. schedule is recoverable. Time stamp ordering schedule with Thomas write rule generates view serial schedule with blind write. Because of blind write, it won't be conflict serial.

6) Let the F.D $F = \{QR \rightarrow S, R \rightarrow P, S \rightarrow Q\}$ hold on a relation schema where $X \rightarrow PQRS$. X is not in BCNF. Suppose X is decomposed into 2 schemas $Y = PR$ and $Z = QRS$. Comment on decomposition. Is $Y \in Z$ in BCNF?

Sol: $X(PQRS) = \{QR \rightarrow S, R \rightarrow P, S \rightarrow Q\}$ is decomposed into $Y(PR) \in Z(QRS)$.

\Rightarrow $Y(PR)$ $Z(QRS)$
 $R \rightarrow P$ $QR \rightarrow S$
 $S \rightarrow Q$

Candidate key: R

Relation Y is in BCNF ~~for every schema~~

Candidate key: $\{QR, RS\}$

Relation Z is in 3NF but not BCNF ($\because S$ is not superkey).

7) Consider the schema with non-trivial F.D

(i) Registration (rollno, counsid, email)

(ii) rollno, counsid \rightarrow email

(iii) email \rightarrow rollno.

Justify whether schema is in 3NF & in BCNF.

Sol: Since $\{\text{rollno, counsid}\}$ is primary key so rollno & counsid are prime attributes. email is non-prime attribute.

FD rollno, counsid \rightarrow email is in BCNF & 3NF but FD email \rightarrow rollno violates the rule of BCNF because email is not superkey. But it satisfies rule of 3NF because rollno is prime \rightarrow attribute.

So, overall this relation is in 3NF but not in BCNF.

- 8) A database of research articles in a journal uses the following schema (Volume, Number, Startpage, Endpage, title, year, price). The primary key is (Volume, Number, Startpage, Endpage) & no following FD exist in the schema. (Volume, Number, Startpage, Endpage) \rightarrow Title (Volume, Number \rightarrow Year (Volume, Number, Startpage, Endpage) \rightarrow Price). The database is redesigned to use the following schema. (Volume, Number, Startpage, Endpage, title, Price) (Volume, Number, year). Which is the weakest normal form that the new database satisfies, but old one does not?

Sol: Old relation has functional dependency: Volume, Number \rightarrow Year as partial dependency. So it does not follow 2NF. But, there is no partial dependency in the new relation & so it satisfies 2NF as well as 3NF. Therefore, 2NF is the weakest normal form that the new database satisfies, but old one does not.

- 9) Consider following 2 phase locking protocol. Suppose a transaction T accesses (for read or write operations), a certain set of objects $\{O_1, \dots, O_k\}$. Consider the following 2PL protocol. Suppose a transaction T accesses (for read/write operations), a certain set of objects O. It is done in following manner:
 Step 1: T acquires exclusive locks to O_1, \dots, O_k in increasing order of their addresses.
 Step 2: The read operations are performed.
 Step 3: All locks are released k.

~~This is done in following manner:~~ Can deadlock occur? Does transaction achieve serializability.

Sol: The above transaction scenario is conservative 2PL. In conservative 2PL protocol, a transaction has to lock all the items it access before the transaction begins execution. It is used to avoid deadlocks. Also, 2PL is conflict serializable, therefore it guarantees serializability.

- 10) Consider a schedule of transactions T_1 & T_2 :
- | | | | | | |
|-------|----|----|----|----|--------|
| T_1 | RA | RC | WD | WB | Commit |
| T_2 | RB | WB | RD | WC | commit |
- Now RX stands for Read (x) & WX stands for Write (x). With its conflict equivalent schedule.

Sol:

T_1	RA	RC	WD	WB	Commit
T_2	RB	WB	RD	WC	commit

2 transactions T_1 & T_2 are given as

$T_1: r_1(X), w_1(X), r_1(Y), w_1(Y)$

$T_2: r_2(Y), w_2(Y), r_2(Z), w_2(Z)$

Find the total no. of conflict serializable schedules formed by T_1 & T_2 .

In T_1 , conflicting operations $\Rightarrow r_1(Y)$ & $w_1(Y)$

In T_2 , conflicting operations $\Rightarrow r_2(Y)$ & $w_2(Y)$

\therefore only one way for $T_1 \rightarrow T_2$.

$$\text{For } T_2 \rightarrow T_1, \text{ we have } \underbrace{5C1 + 5C2}_{15} + \underbrace{4C1 + 4C2}_{10} + \underbrace{3C1 + 3C2}_{6}$$

$$\Rightarrow 53$$

\therefore total no. of conflict serializable schedules are $53 + 1 = 54$

10) Given a relation $R(P, Q, R, S, T, U, V, W, X, Y, Z)$ & $FD = \{ PQ \rightarrow R, P \rightarrow ST, Q \rightarrow U, U \rightarrow VW, S \rightarrow XY \}$, determine whether given R is in 3NF? If not convert it into 3NF.

Sol:

$$FD = \begin{cases} PQ \rightarrow R \\ P \rightarrow ST \\ Q \rightarrow U \\ U \rightarrow VW \\ S \rightarrow XY \end{cases}$$

R_3 is not in 3NF

$$\Rightarrow R_3'(QU), R_3''(UVW)$$

Now this is in 3NF

R_4 is in 3NF

$$PQ^+ = PQRSTUVWXY$$

$$\Rightarrow \text{candidate key} = PQZ$$

$$\text{Prime attribute} = P, Q, Z$$

$$\text{Non-prime attribute} = R, S, T, U, V, W, X, Y$$

$$PQ \rightarrow R, P \rightarrow ST, Q \rightarrow U \text{ violates 2NF}$$

to make it 2NF first, so as to make it 3NF

$$\Rightarrow R_1(PQR), R_2(PSTXY), R_3(QUVW), R_4(Z)$$

Now this is in 2NF

For 3NF, it should be super key

R_1 is in 3NF, with PQ as super key

R_2 is not in 3NF, given $P \rightarrow STXY \Rightarrow R_2'(PST), R_2''(SXY)$

Now R_2' & R_2'' are in 3NF with P & S as super key

$$\therefore \text{decomposed relation is } R_1(PQR), R_2'(PST), R_2''(SXY), R_3'(QU), R_3''(UVW), R_4(Z).$$