COMP9311

DATABASE SYTEMS

SEMESTER 1,2018

ASSIGNMENT-2

SUBMITTED BY

RAGAVENDRAN LAKSHMINARASIMHAN

Z5179974

QUESTION 1:

(1) No! $C \rightarrow J \notin F^+$.

Reason:

$C^+=\{C\}$

$C^+= \{C, D, H\}$

$C^+= \{C, D, H, G\}$

Since $J \not\subset C^+$. Therefore $C \rightarrow J \notin F^+$

(2)

From the given Functional Dependency, we split:

$A \rightarrow B$, $A \rightarrow C$,

$E \rightarrow A$, $E \rightarrow D$,

$BD \rightarrow E$,

$CE \rightarrow D$,

$CE \rightarrow H$,

$H \rightarrow G$,

$EI \rightarrow J$

Since $E \rightarrow A$ and $A \rightarrow C \Rightarrow E \rightarrow C$ (BY TRANSITIVITY)

Since $E \rightarrow C \Rightarrow E \rightarrow D$ and $E \rightarrow H$

$F_{m=}\{A \rightarrow B, A \rightarrow C, E \rightarrow A, E \rightarrow D, E \rightarrow H, H \rightarrow G, EI \rightarrow J, BD \rightarrow G\}$

(3)INITIAL MATRIX

Relation	A	B	C	D	E	G	H	I	J	K
R1 = $\{ABCDE\}$	a	a	a	a	a	b	b	b	b	b
R2 = $\{EGH\}$,	b	b	b	b	a	a	a	b	b	b
R3 = $\{EIJK\}$	b	b	b	b	a	b	b	a	a	a

FINAL MATRIX:

Relation	A	B	C	D	E	G	H	I	J	K
R1 = $\{ABCDE\}$	a	a	a	a	a	b	b	b	b	b
R2 = $\{EGH\}$,	b	b	b	b	a	a	a	b	b	b
R3 = $\{EIJK\}$	a	a	a	a	a	a	a	a	a	a

Matrix after applying all functional dependencies, last row consists of all “a” symbols. Therefore, we stop.

$\therefore$ R1={ABCDE},R2={EGH} and R3={EIJK} are lossless-join.

(4) LIST OF SUPERKEYS:

1. SUPERKEY: {A B C D E G H I J K}
2. SUPERKEY: {A D E G H I J K} (Since A→BC)
3. SUPERKEY: {E G I J K} (Since E→ADH)
4. SUPERKEY: {E I J K} (Since E →ADH and H→G)
5. SUPERKEY: {E I K} (Since EI→J)
6. SUPERKEY: {E I K A}
7. SUPERKEY: {E I K B}
8. SUPERKEY: {E I K C}
9. SUPERKEY: {E I K D}
10. SUPERKEY: {E I K G}
11. SUPERKEY: {E I K A B}
12. SUPERKEY: {E I K A B C}
13. SUPERKEY: {E I K A H}

(5) No! It is not possible to decompose R into collection of BCNF relations and ensure the decomposition is dependency-preserving and lossless-join.

BCNF RELATION, we have:

R_1{ABC}

R_2{EAD}

R_3{HG}

R_4{EIJ}

R_5{EHIK}

$R_1^+ \cup R_2^+ \cup R_3^+ \cup R_4^+ \cup R_5^+ \neq F$ (Since BD→E)

So it is not dependency preserving.

Relation	A	B	C	D	E	G	H	I	J	K
R_1{ABC}	a	a	a							
R_2{EAD}	a			a	a					
R_3{HG}						a	a			
R_4{EIJ}					a			a	a	
R_5{EHIK}	a	a	a	a	a	a	a	a	a	a

Matrix after applying all functional dependencies, last row consists of all “a” symbols. Therefore, we stop.

∴ R1,t2, are lossless-join.

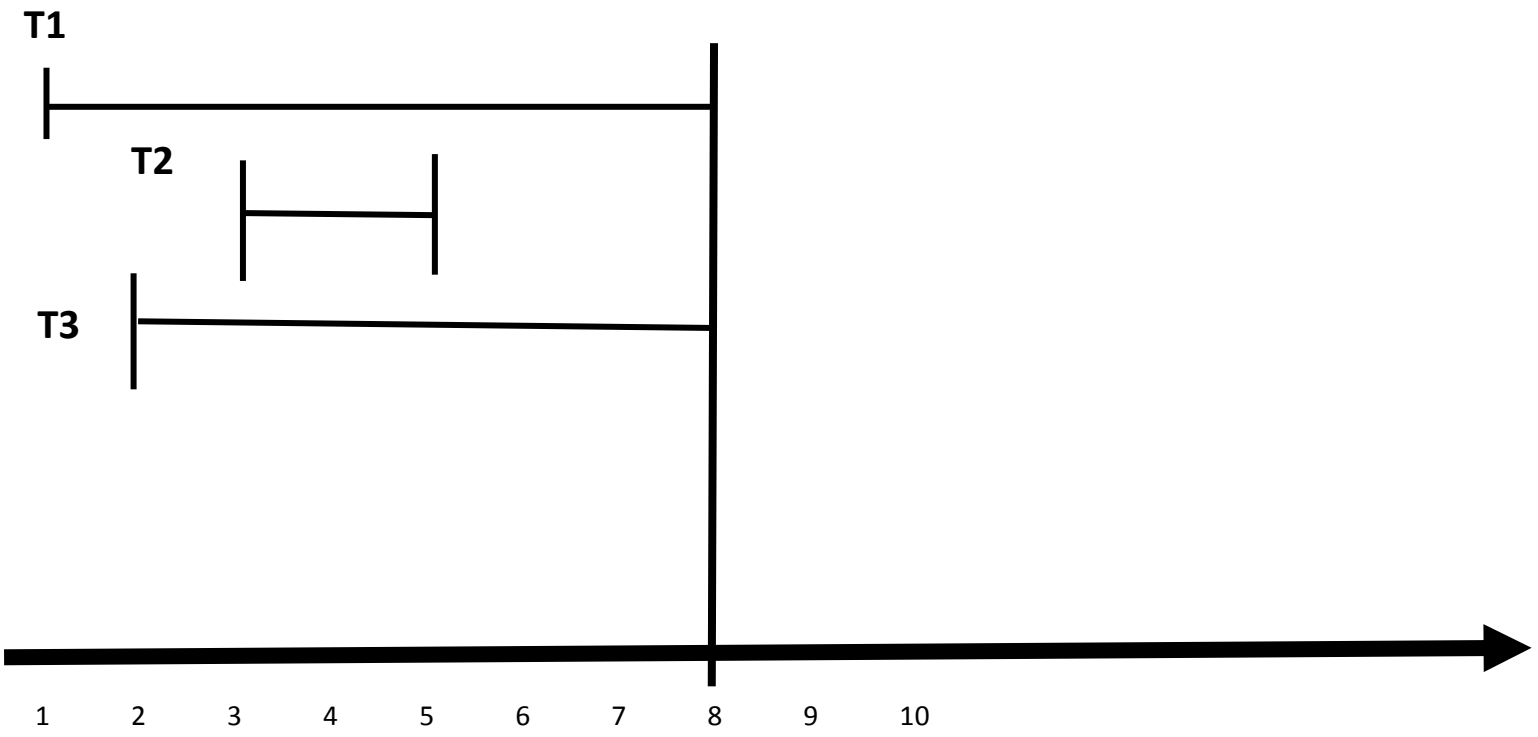
IT IS A LOSSLESS JOIN but not DEPEDENCY PRESERVING.

QUESTION 2:

(1) To recover the system after the crash, we do the following.

UNDO: T1,T3

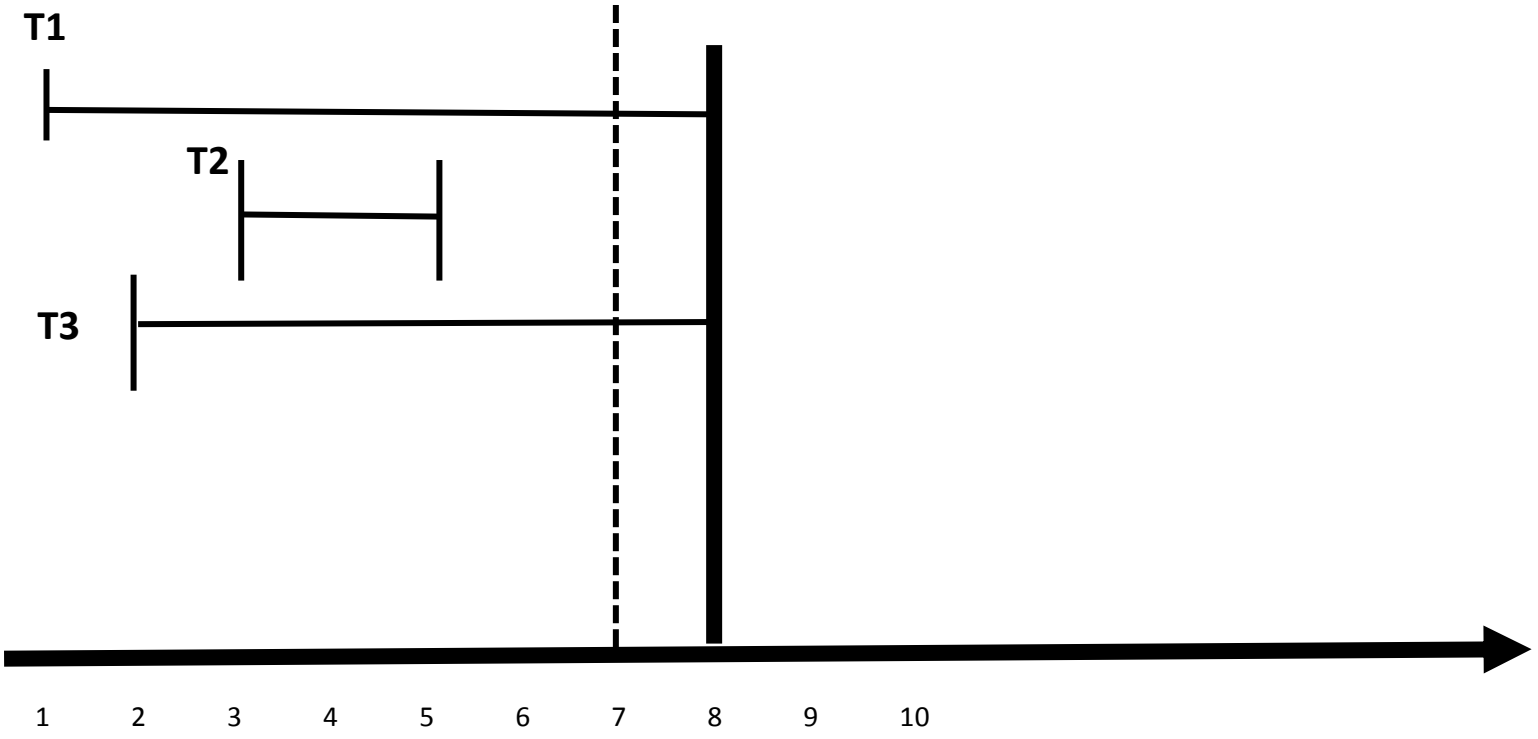
REDO: T2



(2) When the checkpoint is at 7, we need to:

UNDO: T1, T3

NO NEED TO REDO T2 as it is already reflected in the database.

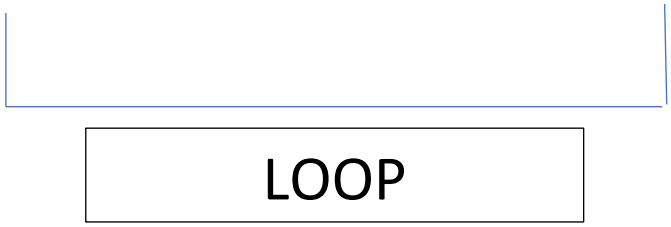


QUESTION 3:

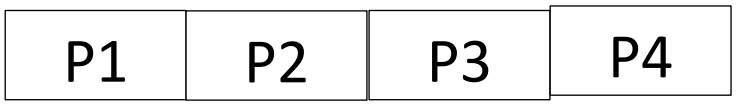
1)Data Pages:1,2,3,4,5

Q1: read P1; Q2:read P2; Q3: read P3 ;Q4:read P4;

Q5:read P5,Q6:read P4,Q7: read P4



Buffer Pages:



In this scenario FIFO is better than MRU.

FIFO:

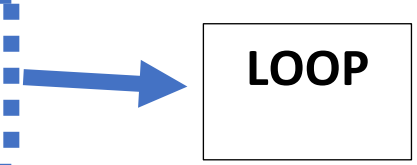
Initially the first four values will be added to the buffer(P1,P2,P3,P4).Then,P1 will be replaced with P5.Again two “4” will be in the buffer. So no replacement required. For every loop buffer replacement is done only once.

MRU:

Initially the first four values will be added to the buffer(P1,P2,P3,P4).Then P4 will be replaced with P5.Again for the two “4”,P5 will be changed two times. Therefore for every loop, buffer replacement is done twice.

2) Data Pages:1,2,3,4,5

Q1: read P1; Q2:read P2;
Q3: read P3 ;Q4:read P1; Q5:read P4,
Q6:read P5,Q7: read P2



Buffer Pages:

P1	P2	P3	P4

In this scenario FIFO is better than MRU.

FIFO:

Initially the first four values will be added to the buffer(P1,P2,P3,P4).Then P1 will be replaced with P5 and P2 will be in the buffer. So only one buffer replacement is done only once.

LRU:

Initially the first four values will be added to the buffer(P1,P2,P3,P4).Then P2 will be replaced with P5(LRU).The next least recently visited P3 will be

changed with P2. Therefore for every loop, buffer replacement is done twice.