COMP9311 ASS2

YIZHENG YING
Z5141180

Q1:

1) $C \to J \notin F+$, because in this case, there is no connection between C and J.

2) $F_{\min} = \{A \to B, A \to C, E \to A, E \to D, BD \to E, E \to H, H \to G, EI \to J\}$

$F = \{\mathrm{A} \to \mathrm{BC}, \mathrm{E} \to \mathrm{AD}, \mathrm{BD} \to \mathrm{E}, \mathrm{CE} \to \mathrm{DH}, \mathrm{H} \to \mathrm{G}, \mathrm{EI} \to J\}$

Step1: So, $F' = \{A \to B, A \to C, E \to A, E \to D, BD \to E, CE \to D, CE \to H, H \to G, EI \to J\}$

Step: Then, check BD→E, $B+ = \{B\}, D+ = \{D\}$, so it can not be replaced.

Then, check CE→D,CE→H, $C+ = \{C\}, E+ = \{EADBCHG\}$, so CE→D can be replaced by E→D and CE→H can be replaced by E→H.

Then, check EI→J, $E+ = \{EADBCHG\}, I+ = \{I\}$, so it can not be replaced.

So, the minimal cover for F is:

$F'' = \{A \to B, A \to C, E \to A, E \to D, BD \to E, E \to H, H \to G, EI \to J\}$

Step3:

$A+ \mid_{F''-\{A \to B\}} = \{A, C\}$; thus, A→B is not redundant.

$A+ \mid_{F''-\{A \to C\}} = \{A, B\}$; thus, A→C is not redundant.

$E+ \mid_{F''-\{E \to A\}} = \{E, D, H, G\}$; thus, E→A is not redundant.

$E+ \mid_{F''-\{E \to D\}} = \{E, A, B, C, H, G\}$; thus, E→D is not redundant.

$BD+|_{F''-\{BD \to E\}} = \{D, B\}$; thus, BD→E is not redundant.

$E+|_{F''-\{E \to H\}} = \{E, A, D, B, C\}$; thus, E→H is not redundant.

$H+|_{F''-\{H \to G\}} = \{H\}$; thus, H→G is not redundant.

$EI+|_{F''-\{EI \to J\}} = \{E, I\}$; thus, EI→J is not redundant.

Thus, $F_{\min} = \{A \to B, A \to C, E \to A, E \to D, BD \to E, E \to H, H \to G, EI \to J\}$

3) It is lossless-join.

Firstly, we can get the table:

	A	B	C	D	E	G	H	I	J	K
R1	a1	a2	a3	a4	a5	b16	b17	b18	b19	b110
R2	b21	b22	b23	b24	a5	a6	a7	b28	b29	b210
R3	b31	b32	b33	b34	a5	b36	b37	a8	a9	a10

For E→AD, R1,R2,R3 are same, so the column A should be changed to a1,column D should be changed to a4.

	A	B	C	D	E	G	H	I	J	K
R1	a1	a2	a3	a4	a5	b16	b17	b18	b19	b110
R2	a1	b22	b23	a4	a5	a6	a7	b28	b29	b210
R3	a1	b32	b33	a4	a5	b36	b37	a8	a9	a10

Considering the A→BC,the value of column A are same, so:

	A	B	C	D	E	G	H	I	J	K
R1	a1	a2	a3	a4	a5	b16	b17	b18	b19	b110
R2	a1	a2	a3	a4	a5	a6	a7	b28	b29	b210
R3	a1	a3	a3	a4	a5	b36	b37	a8	a9	a10

For CE→H, the value of column C and E are same,so:

	A	B	C	D	E	G	H	I	J	K
R1	a1	a2	a3	a4	a5	b16	a7	b18	b19	b110
R2	a1	a2	a3	a4	a5	a6	a7	b28	b29	b210
R3	a1	a3	a3	a4	a5	b36	a7	a8	a9	a10

For H→G , the value of column H are same,so:

	A	B	C	D	E	G	H	I	J	K
R1	a1	a2	a3	a4	a5	a6	a7	b18	b19	b110
R2	a1	a2	a3	a4	a5	a6	a7	b28	b29	b210
R3	a1	a3	a3	a4	a5	a6	a7	a8	a9	a10

	A	B	C	D	E	G	H	I	J	K
R1	a1	a2	a3	a4	a5	a6	a7	b18	b19	b110
R2	a1	a2	a3	a4	a5	a6	a7	b28	b29	b210
R3	a1	a3	a3	a4	a5	a6	a7	a8	a9	a10

As we can see from the table above, the raws of R3 are all of the value aj, so it is lossless-join.

4)

$A+=\{\text{ABC}\}, \text{E}+=\{\text{EADBCHG}\}, \text{BD}+=\{BDEACHG\}, CE+=\{CEDHABG\}, H+=\{HG\}$
$, EI+=\{EIJADBCHG\}$

There is no super key ,so we need to find a new key by using augment rule.

$EIK+=\{EIJADBCHGK\}$

As EIK is a candidate key, so super keys are:

EIK,AEIK,HEIK,CEIK,BDEIK

5)No, it is not possible.

$EIK+=\{EIKABCDHGJ\}$,EIK is a key

candidate key=EIK

The BCNF violations are: A→BC, E→AD, BD→E, CE→DH, H→G, EI→J

$R=\{\text{ABCDEGHIJK}\}$

$fds=\{\text{A}\rightarrow\text{BC},\text{E}\rightarrow\text{AD},\text{BD}\rightarrow\text{E},\text{CE}\rightarrow\text{DH},\text{H}\rightarrow\text{G},\text{EI}\rightarrow\text{J}\}$

Key=EIK

A→BC, does not have the key on the LHS.

So, decomposite to R0=ABC, R1=ADEGHIJK

For R0: $fds=\{\text{A}\rightarrow\text{BC}\}$, key=A, R0 is in BCNF.

For R1: $fds=\{\text{E}\rightarrow\text{AD},\text{H}\rightarrow\text{G},\text{EI}\rightarrow\text{J}\}$,key=EIK, E→AD does not have the key on the LHS, R1 decomposites to R11=EAD, R12=EGHIJK

For R11: $fds = \{E \to AD\}$, key=E, R11 is in BCNF.

For R12: $fds = \{H \to G, EI \to J\}$,key=EIK, H→G does not have the key on the LHS, so R12 decomposites to R13=HG, R14=EHIJK

For R13, $fds = \{H \to G\}$,key=H, R13 is in BCNF.

For R14, $fds = \{EI \to J\}$, key=EIK, EI→J does not have the key on the LHS,so R14 decomposites to R15=EIJ, R16=EHIK

For R15, $fds = \{EI \to J\}$,key=EI, R15 is in BCNF.

For R16, $fds = \{\}$,key=EIK, so R16 is in BCNF.

So BCNF:=R0,R11,R13,R15,R16=(ABC)(EAD)(HG)(EIJ)(EHIK)

We can get the table as follow:

	A	B	C	D	E	G	H	I	J	K
R0	a1	a2	a3	b14	b15	b16	b17	b18	b19	b110
R11	a1	b22	b23	a4	a5	b26	b27	b28	b29	b210
R13	b31	b33	b33	b34	b35	a6	a7	b38	b39	b310
R15	b41	b42	b43	b44	a5	b46	b47	a8	a9	b410
R16	b51	b52	b53	b54	a5	b56	a7	a8	b59	a10

And then we can get the new table:

	A	B	C	D	E	G	H	I	J	K
R0	a1	a2	a3	b14	b15	b16	b17	b18	b19	b110
R11	a1	a2	a3	a4	a5	a6	a7	b28	b29	b210
R13	b31	b33	b33	b34	b35	a6	a7	b38	b39	b310
R15	a1	a2	a3	a4	a5	a6	a7	a8	a9	b410
R16	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10

So for all rows in R16 are value of a.

However, $R0 \cup R11 \cup R13 \cup R15 \cup R16$ is a subset of R, so it is not dependency-preserving, so it is not possible to decompose R into a collection of BCNF relations.

Q2:

1) From the time 8 ,T2 has been committed and T1,T3 has not, so we should redo T2 and undo T1,T3.

2) T2 has been committed,so T2 do not need to redo;

T1,T3 have not been committed, so undo T1, T3.

Q3:

1)

FIFO is better than MRU

For example, let the buffer capacity be 3, let pages be {1,3,2,5,2,2,2}

For FIFO:

PAGES	1	3	2	5	2	2	2
	1	3	2	5	5	5	5
		1	3	3	3	3	3
			1	2	2	2	2
Memory	+1	+1	+1	+1			

So the memory request is 4.

For MRU:

PAGES	1	3	2	5	2	2	2
	1	3	2	5	2	2	2
		1	3	3	3	3	3
			1	1	1	1	1
Memory	+1	+1	+1	+1	+1		

So the memory request is 5, which is more than that FIFO requests.

2)

FIFO is better than LRU

For example, let the buffer capacity be 3, let pages be {1,3,1,5,7,3}

For FIFO:

PAGES	1	3	1	5	7	3
	1	3	3	5	7	7

		1	1	3	5	5
				1	3	3
Memory	+1	+1		+1	+1	

So the memory request is 4.

For LRU:

PAGES	1	3	1	5	7	3
	1	3	3	5	5	5
		1	1	3	7	7
				1	1	3
Memory	+1	+1		+1	+1	+1

So the memory request is 5, which is more than that FIFO requests.