A3

Jing Huang, 1003490705, huang750
Si Tong Liu, 1004339628, liusi17

November 2019

Question 1:

1. Assume $C : course, P : professor, A : teaching\ assistant, L : location, D : day, T : time$

(1) $CDT \rightarrow P$

(2) $C \rightarrow A$

(3) $LDT \rightarrow C$

(4) $LDT \rightarrow P$

(5) $CDT \rightarrow PC$

2. Firstly, do the closure test. Given the FDs in part a) ,we can get

C^+	{CA}
LDT^+	{LDTPCA}
CDT^+	{CDTPA}
A^+	{A}
T^+	{T}
D^+	{D}
L^+	{L}
P^+	{P}

According to the table, we can obtain that the superkey is LDT. Except $LDT \rightarrow PC$ satisfies 3NF with LHS is a superkey, the rests do not satisfy since neither the LHS is superkey nor the RHS is prime. Therefore, the schema does not satisfies 3NF. Except $LDT \rightarrow PC$ satisfies BCNF with PC is not a subset of LDT and LDT is a superkey, the rests do not satisfy since none of them has LHS superkey and is non-trivial. Therefore, the schema does not satisfies BCNF.

In conclusion, this is a bad design.

One better solution is:

Assume Schedule1(L, D, T, P). According to $LDT \rightarrow P$, LDT is a superkey so BCNF holds.

Assume Schedule2(L, D, T, A, C), since $C \to A$, we can decompose this into Schedule3(C, A) and Schedule4(L, D, T, C). According to $LDT \to C$, LDT is a superkey, so BCNF holds.

Therefore, by using BCNF decomposition, we decompose the schema into (L, D, T, P) and (A, C) and (L, D, T, C)

Question 2:
1. $A \to B$, $C \to B$, $AC \to B$
2. There would be no change. Since changing c_3 to c_2 does not make C unique and there are still two different values c_1 to c_2 for a_2, the trivial FDs do not change. For non-trivial FDs, BC is still not unique and AC is still uniqe, so non-trivial FDs do not change as well.

Question 3:
(a)

A^+	{AB}
B^+	{B}
C^+	{ACGDHEBD}
D^+	{D}
E^+	{EF}
F^+	{F}
G^+	{G}
H^+	{AEHBDFGC}

$C->ACG \wedge C->AFH => C->ACGFH$ by Union.

Therefore, the candidate keys would be C and H

(b) i) True. Firstly, find the minimal basis of R.
step (1) : $\{A \to B, ABD \to F, ADB \to G, ABD \to H, AEH \to B, AEH \to D, BC \to E, BC \to H, C \to A, C \to G, C \to F, C \to H, DE \to H, DE \to B, DF \to A, DF \to C, E \to F, H \to E, H \to A\}$

step (2): $A \to B$ cannot be removed since no other LHS for A

$ABD \to F$ can be removed since $ABD^+ = \{ABDHEFCG\}$,F is inside

$ABD \to G$ can be removed since $ABD^+ = \{ABDHEFCG\}$,G is inside

$ABD \to H$ cannot be removed since no other LHS for ABD

$AEH \to B$ can be removed since $AEH^+ = \{AEHBFCGD\}$, B is inside

$AEH \to D$ cannot be removed since no other LHS for AEH

$BC \to E$ can be removed since $BC^+ = \{BCHAGFE\}$, E is inside

$BC \to H$ cannot be removed since no other LHS for BC

$C \to A$ can be removed since $C^+ = \{CGFHEABD\}$ A is inside

$C \to G$ cannot be removed since $C^+ = \{CFHEABD\}$, G is not inside

$C \to F$ can be removed since $C^+ = \{CGHEFABD\}$, F is inside

$C \to H$ cannot be removed since $C^+ = \{CG\}$, H is not inside

$DE \to H$ can be removed since $DE^+ = \{DEBFCGHA\}$, H is inside

$DE \to B$ cannot be removed since no LHS for DE

$DF \to A$ can be removed since $DF^+ = \{DFCGHEABF\}$, A is inside

$E \to F$ cannot be removed since no other LHS for E

$H \to E$ cannot be removed since $H^+ = \{HAB\}$, H is not inside

$H \to A$ cannot be removed since $H^+ = \{HEF\}$, H is not inside

Therefore, the minimal bais is $\{A \to B, ABD \to H, AEH \to D, BC \to H, C \to G, C \to H, DE \to B, DF \to C, E \to F, H \to E, H \to A\}$

The final Fds would be $\{A \to B, ABD \to H, AEH \to D, BC \to H, C \to GH, DF \to C, DE \to BF, H \to EA\}$

By using Union axiom on $C \to G$ and $C \to H$, we can get $C \to GH$. By using Union axiom on $H \to E$ and $H \to A$, we can get $H \to EA$. By using Augmentation axiom, $E \to F$ becomes $DE \to DF$. Since D is a subset of DE, D can be got rid of. Then by using Union axiom, $DE \to F$ and $DE \to B$ become $DE \to BF$.

We can find the closure of C that $C^+ = \{CGHEABDF\}$ which means C is a candidate key. And $H^+ = \{HEABDFCG\}$. Therefore, the a 3NF decomposition result in the same candidate keys.

ii) False, according to part i), after removing the unnecessary FDs, wen can get the final FDs are:
we can decompose the Relation into $\{ABDH\}$,$\{AEHD\}$,$\{BCH\}$
,$\{CGH\}$, $\{DEBF\}$,$\{DFC\}$
or$\{AB\}$,$\{ABDH\}$,$\{AEHD\}$,$\{BCH\}$,$\{CG\}$,$\{CH\}$,$\{DEB\}$,$\{DFC\}$,$\{EF\}$,$\{HE\}$,
$\{HA\}$

There could be multiple ways to decompose this.

Question 4

	Serializable	Confilct-Serializable	View-Serializable	Recoverable	Avoids Cascading Aborts
1	✓	✓	✓	?	?
2	✓	✓	✓	✓	✓
3	✓	✓	✓	X	X
4	✓	✓	✓	✓	✓
5	X	X	X	✓	✓

Reason for the "?" in the number 1 schedule: We do not know whether T_1 and T_2 have commit or abort. So we are not sure whether it is recoverable or ACA.

Question 5

2. The output of Session B is different from Session A. Since the transaction that inserting a new record to Accounts in Session A is not committed. The possible solution is to commit the insert in Session A.

4.
username name balance
capt Captain America 1250.00
hulk The Hulk 4750.00
iron Iron Man 6000.00.00
nat Black Widow 550.00
thor Thor 4250.00

7.
username name balance
capt Captain America 1250.00
hulk The Hulk 5500.00
iron Iron Man 6000.00.00
nat Black Widow 550.00
thor Thor 3500.00

It is ok to transfer monery from "Black Widow" 's account. But transfering money to Hulk's account can not be executed. In mysql server, it is in a waiting state. The reason is that in Session A, the "autocommit" is off and "READ COMMITTED", shared locks prevent transactions in Session B to modify Hulk's balance which is under transaction in Session A and is not committed

8.
Hulk's balance is now: 5800.00

11.
username name balance
capt Captain America 8890.00
hulk The Hulk 1160.00
iron Iron Man 3000.00.00
nat Black Widow 250.00
thor Thor 3500.00

The "Caption America"'s balance is 8890. It reflects the latest transfer from Iron Man. For "READ COMMITTED", it does not allow other transactions to read the modified data that is not yet committed. The Read Committed isolation level prevents the Dirty Read. But for "READ UNCOMMITTED", it allows other transactions to read the modified data which is not yet committed. And it allows the Dirty Read. For "REPEATABLE READ", no other transaction can modify the same data that the current transaction until the current transaction completes.

13.
username name balance
capt Captain America 5890.00
hulk The Hulk 1160.00
iron Iron Man 3000.00.00
nat Black Widow 250.00
thor Thor 3500.00