CS 353
Section 2
Homework 4

Muhammad Arham Khan
21701848

Q1) (a) Yes, this functional dependency holds

(b) No, this functional dependency does not hold because of two scenarios:

(1) C in T_1 = C in T_3 = c1 but the value of A in $T_1 \neq$ A in T_3.
(2) C in T_2 = C in T_4 = c2 but the value of A in $T_2 \neq$ A in T_4.

(c) No because this functional dependency does not hold because:

(1) A in T_1 = A in T_2 = a1 but the value of B in $T_1 \neq$ B in T_2

(d) No, this functional dependency does not hold because of two scenarios

(1) C of T_1 = C of T_3 = c1 but B of $T_1 \neq$ B of T_3
(2) C of T_2 = C of T_4 = c2 but B of $T_2 \neq$ B of T_4

(e) No, this functional dependency does not hold because of two scenarios:

(1) A of T_3 = A of T_4 = a2 but C of $T_3 \neq$ C of T_4
(2) A of T_1 = A of T_2 = a1 but C of $T_1 \neq$ C of T_2

(f) No, this functional dependency does not hold because
B of T_3 = B of T_4 = b3 but C of $T_3 \neq$ C of T_4

(g) Yes, this functional dependency holds

(h) Yes, this functional dependency holds

(i) No, this functional dependency does not hold because
A of T_3 = A of T_4 = a2. but C of $T_3 \neq$ C of T_4

Q2 $P(A, B, C, D)$
$Q(A, B, C, D)$

f.d for $P = (A \rightarrow BCD,\ B \rightarrow ACD)$
f.d for $Q = (BC \rightarrow AD,\ D \rightarrow B)$

(a) (1) Attributes in relation only = None
(2). Attributes in right side of f.d = CD
(3) Attributes in left side of f.d = None

$\rightarrow P - (1) - (2) - (3) = ABCD - CD$
$= AB$

$\rightarrow$ find $\{A\}^+$ and $\{B\}^+$ now

$\{A\}^+ \Rightarrow$ result = A
as $A \rightarrow BCD$ and result contains A
result = ABCD
as reult = P So, A is candidate key

Hence :- A and B are candidate keys here

(b) (1). Attributes in relation only = None
(2) - Attributes in right side of f.d = A
(3) . Attributes in left side of f.d = C

$\rightarrow Q - (1) - (2) - (3) = ABCD - AC = BD$

$\{BC\}^+ \rightarrow$ result = BC
as $BC \rightarrow AD$ and result = ABCD
so BC is also a candidate key

$\{CD\}^+ \rightarrow$
result = CD
as $D \rightarrow B$ and result = BCD
so $BC \rightarrow AD$ and result = ABCD
So BD is a also a candidate key

Hence: $\rightarrow$ BC and CD are candidate keys for Q.

Scanned with CamScanner

Q3 $S(A, B, C, D, E, F, G)$

. First we will find the canonical cover for S

$\{B\}^+ = B$

$\{C\}^+ = CD$ } Since D is redundant, it alters the result

$\{D\}^+ = D$

$\{BC\}^+ = ABCDEFG$

$\{BD\}^+ = BD$

$\{CD\}^+ = CD$

$F = \{BC \to A,\ BC \to E,\ A \to F,\ F \to G,\ C \to D,\ A \to G\}$

Now we will remove transitive dependency

$F = \{BC \to A,\ BC \to E,\ A \to F,\ F \to G,\ C \to D\}$

Now we try to find candidate Key as in Q2.

---

1 - Attributes in relation only = None

2 - Attributes in right side of f.d = EGD

3 - Attributes in left side of f.d = BC

$\to S - 1 - 2 - 3 = ABCDEFG - EGDBC = BC$

$\{BC\}^+ = ABCDEFG$ as a result

Hence BC is candidate key of S

2N.F (removing partial dependencies)

As $C \to D$ is a partial dependency, we create a new relation out of it

So, $S_1(A, B, C, E, F, G)$ and $S_2(C, D)$

Candidate key of S_1 = BC

F.d of $S_1 = \{BC \to A,\ BC \to E,\ A \to F,\ F \to G\}$

Candidate key of S_2 = C

F.d of $S_2 = \{C \to D\}$

3N.F (removing all non-prime dependencies)

non-prime attributes are :-

$S_1 = S_1 - BC = AEFG$

$S_2 = S_2 - C = D$

Now we will remove all non-prime dependancy by making new relations out of them.

So,

$\left\{\begin{array}{l} S_1\ (A, B, C, Z) \\ S_2\ (A, F) \\ S_3\ (F, G) \\ S_4\ (C, D) \end{array}\right\} \rightarrow$ the relations in 3.N.F

Q4) (a) $Q(A, B, C, D, E)$

F.d for $Q = \{AB \rightarrow E, D \rightarrow C\}$

1. Attributes in relation only : None
2. Attributes in right side of f.d: CE
3. Attributes in left side of f.d: ABD

$\{ABD\}^+ = ABCDE$ as a result and so,

ABD is a candidate key and any and all subset of Q that contains ABD is now a super key.

ABD, ABCD, ABDE, ABCDE are the 4 super keys

(b) To begin with,

F.d of the original relations are the same as the canonical cover. ie: $\{AB \rightarrow E, D \rightarrow C\}$

Then we will decompose into partial dependencies to satisfy 2.N.F

$AB \rightarrow E$ and $D \rightarrow C$ are both partial dependencies.

$Q_1 (A, B, E)$ candidate key: AB

$Q_2 (D, C)$ candidate key: D

$Q_3 (A, B, D)$ candidate key: ABD

- So now since theres no transitive dependencies its in 3N.F
- Since there is no functional dependency which nipon left side so,

$Q_1 (AB, E)$ $Q_2 (Q, C)$ $Q_3 (ABD)$ are the BCNF decomposition.

Scanned with CamScanner

Q5

(a) $S(A,B,C,D,E)$ $F = \{A \to C,\ BD \to A,\ D \to E\}$
the decompositions for this will be : $S_1(B,C,D)$, $S_2(A,B,D)$ and $S_3(A,E)$

So for lossless join,

1- $S_1 \cap S_2 \cap S_3 \neq \emptyset$

So, as $BCD \cap ABD \cap AE = \emptyset$ and hence the decomposition is not lossless.

(b) $S(A,B,C,D)$ $F = \{A \to BCD,\ B \to C,\ CD \to A\}$
the decompositions for this will be : $S_1(A,B,C)$ and $S_2(B,C,D)$

1- $S_1 \cap S_2 = (ABC) \cap (BCD) = BC \neq 0$
2- $S_1 \cup S_2 = (ABC) \cup (BCD) = ABCD = S$
3- $S_1 \cap S_2 = BC \to S_1$ or S_2

Because $\{BC\}^+ = BC$ and $BC \not\to ABC$, $BC \not\to BCD$ and so the decomposition will not be lossless.

(c) $S(A,B,C,D)$ $F = \{A \to BCD,\ B \to C,\ CD \to A\}$ and the decomposition for this will be : $S_1(A,B,D)$, $S_2(B,C)$

1- $S_1 \cap S_2 = ABD \cap BC = B \neq \emptyset$
2- $S_1 \cup S_2 = ABD \cup BC = ABCD = S$
3- $S_1 \cap S_2 = B \to S_1$ or S_2

Because $\{B\}^+ = BC$ and then $B \to BC$ and as BC is S_2
$B \to S2$ and hence the decomposition will be lossless

Scanned with CamScanner

(Q6) $S(A, B, C, D)$ and F.d of $S = \{A \rightarrow BCD, B \rightarrow C, CD \rightarrow A\}$
and the decompositions for this f.d ~~of~~ relation will be:
$S_1(A, B, C)$ and $S_2(B, C, D)$

So now for every functional dependency we checks if a
a partial dependency does not exist and $\alpha \rightarrow B$

(1) $A \rightarrow BCD$
result will be A
and as $S_1 = ABC$
$S_1 \cap A = A$
$\{S_1 \cap A\}^+ = ABCD$
$\cancel{X} = S_1 \cap \{A \cap S1\}^+ = ABC$
so result $= X \cup$ result $= ABC$

(2) Now since $S_2 = BCD$
$S_2 \cap ABC = BC$
$\{S_2 \cap ABC\}^+ = BC$
$X = S_2 \cap \{ABC \cap S_2\}^+ = BC$
So result $= \cancel{X} \cup$ result $= ABC$

and since the result changed from A to ABC

$S_1 \cap ABC = ABC$
$X = \{ABC \cap S_1\}^+ \cap S_1 = ABC$
result $=$ result $\cup t = ABC$

Now since $ABC \neq BCD$ and $A \rightarrow BCD$ is not in S_1 or S_2
and $CD \rightarrow A$ is not in S_1 or S_2
and $B \rightarrow C$ is in S_2
and so the
decomposition is not dependency preserving.