CS 353 Spring 2020
Homework 4 Solutions
Due: 13 April, Monday till midnight

Q.1 [18 pts, 2 pts each] Given the following instance of the relation R(A,B,C)

	A	B	C
Tuple 1	a1	b1	c1
Tuple 2	a1	b2	c2
Tuple 3	a2	b3	c1
Tuple 4	a2	b3	c2

Which of the following dependencies may hold in relation R? If a dependency cannot hold, indicate which tuples cause the violation.

(a) B → A (b) C → A (c) A → B (d) C → B (e) A→ C (f) B → C (g) BC → A
(h) AC → B (i) AB → C

Answer:

	FD	Holds ?
a)	$B \rightarrow A$	holds
b)	$C \rightarrow A$	fails: tuples 1,3
c)	$A \rightarrow B$	fails: tuples 1,2
d)	$C \rightarrow B$	fails: tuples 1,3
e)	$A \rightarrow C$	fails: tuples 1,2
f)	$B \rightarrow C$	fails: tuples 3,4
g)	$BC \rightarrow A$	holds
h)	$AC \rightarrow B$	holds
i)	$AB \rightarrow C$	fails: tuples 3,4

Q.2 [8 pts, 4 pts each] Given the following relations with functional dependencies:
Relations: P(A, B, C, D) and Q(A, B, C, D)
Functional Dependencies:
- For P: A→BCD, B→ACD
- For Q: BC → AD, D→B

(a) Find all candidate keys in P
(b) Find all candidate keys in Q

Answer:
a) A and B
b) BC and CD

Q.3 [20 pts] Given the following relation with functional dependencies:
Relation: S(A, B, C, D, E, F, G)
Functional Dependencies: F = {BCD → A, BC → E, A → F, F → G, C→D, A→G}
Decompose S into 3NF. The decomposition should be lossless-join and dependency preserving. Show all your work.

Answer:

The candidate key of this relation is BC. BC is unique ($BC^+ = S$) and minimal ($B^+ = B$, $C^+ = CD$).
Finding the canonical cover:
We apply union rule
F' = {BCD → A, BC → E, A →FG, F → G, C → D}
D is extraneous in BCD → A , Since A is in $(BC)^+$, we replace BCD → A by BC → A
F' = {BC → A, BC → E, A →FG, F → G, C → D}
We apply union rule again
F' = {BC → AE, A →FG, F → G, C → D}
Now consider BC → AE
- Check if A is extraneous in BC → AE:
F'' = {BC → E, A →FG, F → G, C → D}
BC^+ under F'' = BCED does not include A so A is not extraneous.
- Check if E is extraneous in BC → AE:
F'' = {BC → A, A →FG, F → G, C → D}
BC^+ under F'' = BCAD does not include E so E is not extraneous.
Now consider A → FG
- Check if F is extraneous in A → FG:
F'' = {BC → AE, A →G, F → G, C → D}
A^+ under F'' = AG does not include F so F is not extraneous.
- Check if G is extraneous in A → FG:
F'' = {BC → AE, A →F, F → G, C → D}
A^+ under F'' = AFG include G so G is extraneous.
Thus, the canonical cover F' = {BC → AE, A →F, F → G, C → D}
For each X → Y Є F', create a relation:
S1(B,C, A, E), S2(A, F), S3(F, G), S4(C, D)
The candidate key is in S1, and none of the relations is a subset of any other relation.
Therefore, we do not add any more relations.
S is decomposed into S1, S2, S3 and S4 which are all in 3NF.

Q.4 [24 pts] Given the following relation with functional dependencies:
Relation: Q(A,B,C,D,E)
Functional Dependencies: AB → E and D → C
(a) [6 pts] Find all superkey(s) and candidate key(s)
(b) [18 pts] Decompose Q into BCNF. Show all your work.

Answer:

a) A superkey is a set of attributes X s.t. X^+ = all attributes. From the FDs above, we can derive:
$\{A, B, D\}^+ = \{A, B, C, D\}^+ = \{A, B, D, E\}^+ = \{A, B, C, D, E\}^+ = \{A, B, C, D, E\}$
Hence, {A, B, D}, {A, B, C, D}, {A, B, D, E} and {A, B, C, D, E} are all superkeys.
A candidate key is a minimal superkey. {A, B, D} is the only candidate key.

b) Both functional dependencies violate BCNF. Try $\{A, B\}^+ = \{A, B, E\}$. Decompose into Q1(A, B, E) and Q2(A, B, C, D). For Q1, AB → E is the only FD and {A, B} is a candidate key for Q1, so Q1 is in BCNF. Q2 is not in the BCNF, since {D} is not a key and we have D → C.
Try $\{D\}^+ = \{C, D\}$. Decompose into Q3(C, D) and Q4(A, B, D). D is a candidate key for Q3 (D+ = CD), so Q3 is in BCNF. There is no nontrivial FD for Q4, so Q4 is also in BCNF.
As a result, Q is replaced by: Q1(A, B, E), Q3(C, D) and Q4(A, B, D) which are all in BCNF.

Q.5 [18 pts, 6 pts each] Determine if the following decompositions are lossless-join or not. Show all your work.

(a) Given the relational schema S(A, B, C, D, E), and the functional dependency set: { A → C, BD → A, D → E }. S is decomposed into S1(B, C, D), S2(A, B, D) and S3(A, E)

(b) Given the relational schema S(A, B, C, D), and the functional dependency set: { A → BCD, B → C, CD → A }. S is decomposed into S1(A, B, C) and S2(B, C, D)

(c) Given the relational schema S(A, B, C, D), and the functional dependency set: { A → BCD, B → C, CD → A }. S is decomposed into S1(A, B, D) and S2(B, C)

Answer:

a) Decomposition of S into S1(B, C, D), S2(A, B, D) and S3(A, E) is not lossless-join, because S2 ∩ S3 = A is not a superkey for S2 or S3 (neither A → S2 nor A → S3 is true).

b) Decomposition of S into S1(A, B, C) and S2(B, C, D) is not lossless-join, because S1 ∩ S2 = BC is not a superkey for S1 or S2 (neither BC → S1 nor BC → S2 is true).

c) Decomposition of S into S1(A, B, D) and S2(B, C) is lossless-join, because S1 ∩ S2 = B is a superkey for S2.

Q.6 [12 pts]] Given the relational schema S(A, B, C, D), and the functional dependency set: { A → BCD, B → C, CD → A }. Determine if the decomposition of S into S1(A, B, C) and S2(B, C, D) is dependency preserving or not. Show all your work.

Answer:

Decomposition of S into S1(A, B, C) and S2(B, C, D) is not dependency preserving, because A → BCD is lost.

- For A → BCD we have:
result = A
using S1:
$(\text{result} \cap S1)^+ = A^+ = ABCD$
$t = (\text{result} \cap S1)^+ \cap S1 = ABCD \cap S1 = ABC$
result = result U t = ABC
using S2:
$(\text{result} \cap S2)^+ = (BC)^+ = BC$
$t = (\text{result} \cap S2)^+ \cap S2 = BC \cap S2 = BC$
result = result U t = ABC

In the second turn of the while loop of the algorithm the result does not change, so we stop here. As the rhs of FD A → BCD is not included in the result, we conclude that the decomposition is not dependency preserving.