CS 1555 – DATABASE MANAGEMENT SYSTEMS (FALL 2018)
DEPT. OF COMPUTER SCIENCE, UNIVERSITY OF PITTSBURGH

Assignment #3: Normalization
Sample Solution

Release: Sept. 20, 2018 Due: 8:00 PM, Wednesday, Sept. 26, 2018

Goal

The goal of this assignment is to understand and gain familiarity with conceptual database design. You will practice using normalization techniques.

Answer the following questions [for a total of 100 points]:

1. Consider the following set of functional dependencies:

   FD1: ProductID → Length, Width, Height, Weight

   FD2: OrderID → OrderDate, CustomerID, TotalPrice

   FD3: CustomerID → Address, City, State, ZipCode, PhoneNumber

   FD4: ProductID, OrderID → ProductQuantity

   (a) [30 points] Using universal relational approach (top-down process), construct a set of 3NF/BCNF relations from the above functional dependencies. Indicate the primary keys for the result relations. Please show all steps clearly as mentioned in the lecture slides.

   (b) [20 points] Using the table method, check whether the constructed set of relations is lossless or not. Also, state if your decomposition is good, bad or ugly.
   Hint: Bad decomposition is a lossy one, while ugly decomposition is lossless but does not preserve some dependencies.

   Solution to (a)

   (a) No need to eliminate any FD since the set represents the canonical cover.

   (b) We construct the universal relation (UR) by taking the union of the attributes in the FDs. This means that all FDs are localized within UR.
   UR= (ProductID, Length, Width, Height, Weight, OrderID, OrderDate, CustomerID, TotalPrice, CustomerID, Address, City, State, ZipCode, PhoneNumber, ProductQuantity)

   (c) Find the primary key of UR. Observations:

   i. ProductID and OrderID do not appear on the right hand side of any FD, so they have to appear in all keys of R.

   ii. (ProductID,OrderID)+ :
   (ProductID,OrderID) → (ProductID,OrderID, ProductQuantity) [from FD4]
   → (ProductID,OrderID, ProductQuantity, OrderDate, CustomerID, TotalPrice) [from FD2]

→ (ProductID,OrderID, ProductQuantity, OrderDate, CustomerID, TotalPrice, Address, City, State, ZipCode, PhoneNumber) [from FD3]
→ (ProductID,OrderID, ProductQuantity, OrderDate, CustomerID, TotalPrice, Address, City, State, ZipCode, PhoneNumber, Length, Width, Height, Weight) [from FD1].

So (ProductID,OrderID) is a key of UR.
In this case, we do not need to consider any other combination, because any other combination containing (ProductID,OrderID) is a super key and not minimal.

(d) UR is in 1NF but not in 2NF since it contains partial dependencies due to FD1 and FD2. So we split UR into three relations along the lines of FD1 and FD2 so that FD1 and FD2 are localized.
R1 = (ProductID, Length, Width, Height, Weight)
R2 = (OrderID, OrderDate, CustomerID, TotalPrice)
R3 = (ProductID, OrderID, ProductQuantity, CustomerID, Address, City, State, ZipCode, PhoneNumber)

(e) R1 and R2 in BCNF but R3 is in 2NF since there is a transitive dependency due to FD3.

(f) Split R3 along the lines of FD3 and the final set of relations which are all BCNF are:
R1 = (ProductID, Length, Width, Height, Weight)
R2 = (OrderID, OrderDate, CustomerID, TotalPrice)
R4 = (CustomerID, Address, City, State, ZipCode, PhoneNumber)
R5 = (ProductID, OrderID, ProductQuantity)

Solution to (b)

Let the attributes be sorted in the following order:
(1) ProductID, (2) Length, (3) Width, (4) Height, (5) Weight, (6) OrderID, (7) OrderDate, (8) CustomerID, (9) TotalPrice, (10) Address, (11) City, (12) State, (13) ZipCode, (14) PhoneNumber, (15) ProductQuantity

Initially the table looks like the following. Note that the table uses simplified marks. “a” means known cell, and empty cell means “U”.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R1	a	a	a	a	a										
R2						a	a	a	a						
R4								a		a	a	a	a	a	
R5	a					a									a

Using FD1, we can add more “a” marks in the table. New marks are in italic.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R1	a	a	a	a	a										
R2						a	a	a	a						
R4								a		a	a	a	a	a	
R5	a	*a*	*a*	*a*	*a*	a									a

Then we use FD2 to update the table. New marks are in italic.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R1	a	a	a	a	a										
R2						a	a	a	a						
R4								a		a	a	a	a	a	
R5	a	a	a	a	a	a	*a*	*a*	*a*						a

Next, we use FD3 to update the table. New marks are in italic.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R1	a	a	a	a	a										
R2						a	a	a	a	*a*	*a*	*a*	*a*	*a*	
R4								a		a	a	a	a	a	
R5	a	a	a	a	a	a	a	a	a	*a*	*a*	*a*	*a*	*a*	a

Now we have 1 rows filled with mark “a”, so it is a lossless decomposition. Since it preserves all FDs, it is a good decomposition.

2. Consider the following set of functional dependencies:

   $K \rightarrow D$,

   $B \rightarrow C$,

   $KB \rightarrow CD$,

   $K \rightarrow C$,

   $K \rightarrow B$,

   $B \rightarrow CD$,

   $E \rightarrow F$,

   $EF \rightarrow G$

   (a) [40 points] Using synthesis approach (bottom-up process), construct a set of 3NF/BCNF relations from the above functional dependencies. Indicate the primary keys for the result relations and whether or not they are in 3NF or BCNF. Please show all steps clearly as mentioned in the lecture slides.

   (b) [10 points] Using the table method, check whether the constructed set of relations is lossless or not. If not, correct them.

Solution to (a)

(a) Find the canonical cover of the FDs.

i. Transform the FDs so that there is only one literal on the right hand side of each FD:

K $\rightarrow$ D
B $\rightarrow$ C
KB $\rightarrow$ C
KB $\rightarrow$ D
K $\rightarrow$ C
K $\rightarrow$ B
B $\rightarrow$ C
B $\rightarrow$ D
E $\rightarrow$ F
EF $\rightarrow$ G

ii. Remove extraneous attributes on the left hand side:
Clearly, the second B $\rightarrow$ C is extraneous.
K in KB $\rightarrow$ C is also extraneous since B $\rightarrow$ C.
KB $\rightarrow$ D is also extraneous since K $\rightarrow$ D and B $\rightarrow$ D.
F in EF $\rightarrow$ G is also extraneous since E $\rightarrow$ F.
Drop the extraneous dependencies, the set becomes:

K $\rightarrow$ B
B $\rightarrow$ C
B $\rightarrow$ D
K $\rightarrow$ D
K $\rightarrow$ C
E $\rightarrow$ F
E $\rightarrow$ G

iii. Drop transitive (redundant) FDs:
D is transitively dependent on K, since K $\rightarrow$ B and B $\rightarrow$ D.
C is transitively dependent on K, since K $\rightarrow$ B and B $\rightarrow$ C.
Drop these functional dependencies.
The canonical cover is:

K $\rightarrow$ B
B $\rightarrow$ C
B $\rightarrow$ D
E $\rightarrow$ F
E $\rightarrow$ G

(b) Find the candidate keys of the relation.
Observation:
K and E does not appear in any right hand side of the FDs, so K and E must be in all keys of R.
KE$^+$ = KBE (since K $\rightarrow$ B) $\rightarrow$ KBCE (since B $\rightarrow$ C) $\rightarrow$ KBCDE (since B $\rightarrow$ D) =

KBCDEF (since E → F) = KBCDEFG (since E → G), so KE is a minimal superkey (candidate key).
Considering the observations, any set of 2 or more attributes have to contain KE, so they are superkey but not minimal.

(c) Construct a set of 3NF/BCNF relations.
Group together FDs with same determinant and construct a relation for each group. Don't forget to add another relation containing the key. Finally, we get:

R1(<u>K</u>,B)
R2(<u>B</u>,C,D)
R3(<u>E</u>,F,G)
R4(<u>K</u>,<u>E</u>)

R1 is in BCNF because for K → B, K is a superkey of R1.
R2 is in BCNF because for B → C and B → D, B is a superkey of R2.
R3 is in BCNF because for E → F and E → G, E is a superkey of R3.
R4 is in BCNF because it has an empty FD set.

Solution to (b)

(a) Initially, the table looks like below. Known cells are marked with "a", while unknown cells are empty.

	K	B	C	D	E	F	G
R1	a	a					
R2		a	a	a			
R3					a	a	a
R4	a				a		

(b) After using K → B, B → C, B → D, we get:

	K	B	C	D	E	F	G
R1	a	a	a	a			
R2		a	a	a			
R3					a	a	a
R4	a	a	a	a	a		

(c) After using E → F and E → G, we get:

	K	B	C	D	E	F	G
R1	a	a	a	a			
R2		a	a	a			
R3					a	a	a
R4	a	a	a	a	a	a	a

(d) We see that all cells of R4 are known. Therefore the set of relations is lossless.