School of Engineering and Computer Science

SWEN304 **Database System Engineering**

Assignment 3

Due date: 23:59, Monday 17 May

The objective of this assignment is to test your understanding of functional dependencies, normal forms, database normalization. The assignment is worth **10%** of your final grade. It will be marked out of 100.

Submission instructions:

- Submit your assignment in **pdf** via the submission system
- Please make sure to write your **student ID and Name** on your assignment. *Note*: Assignments without IDs and names OR not in **pdf** will incur a deduction of 3 marks.

Question 1. Functional Dependencies and Normal Forms [20 marks]

a) [4 marks] Consider a relation schema $N(R, F)$ where $R = \{A, B, C\}$. Suppose we find the following two tuples in an instance of this relation schema.

A	*B*	*C*
1	*2*	*3*
4	*2*	*3*

Which of the following functional dependencies do definitely **not** hold over the relation schema N? Justify your answer.

1) $C \rightarrow A$

No because a C can be associated with more than one A .

2) $A \rightarrow C$

Yes because an A can be associated with at most one C .

3) $B \rightarrow A$

No because an B can be associated with more than one A .

4) $B \rightarrow C$

Yes because an B can be associated with at most one C .

b) [16 marks] Consider a relation schema $N(R, F)$ where $R = \{A, B, C, D\}$. For each of the following sets F of functional dependencies, determine which normal form (1NF, 2NF, 3NF, BCNF) the relation schema N is in. Justify your answer.

Hint: Note that in all four cases AB is the only key for N.

1) $F = \{AB \rightarrow C, C \rightarrow D\}$

2NF, first of all, it is not 3NF or BCNF because from relation C->D, C isn't a super key and D isn't a prime attribute. It is 2NF because all non-prime attributes are not partially functionally dependent on any key, C is fully dependent on AB and D is not dependent on AB at all so it is 2NF.

2) $F = \{AB \rightarrow D, B \rightarrow C\}$

1NF, first of all, it is not 3NF or BCNF because from relation B->C, B isn't a super key and C isn't a prime attribute. It is not 2NF because B->C is partially functional dependency .

3) $F = \{AB \rightarrow C, AB \rightarrow D\}$

BCNF, Because the left-hand side of each non-trivial functional dependency in F contains the relation schema key,AB.

4) $F = \{ AB \rightarrow CD, C \rightarrow B\}$

3NF, first of all it is not BCNF because from dependency C->B,the left hand side does not contain the relation schema key, but it is 3NF because for dependency AB->CD, AB is a superkey and for dependency C->B, B is a prime attribute so it is 3NF.

Question 2. Candidate Key [5 marks] Consider a relation schema $N(R, F)$ where $R=\{A, B, C, D, E\}$ with the set of functional dependencies

$$F = \{AB \rightarrow C, CE \rightarrow D, A \rightarrow B\}$$

Is AB a candidate key of this relation? Explain your answer. Is AE a candidate key of this relation? Explain your answer.

AB cannot be the candidate key of this relation because from AB we can only get C and from C we can not get D without E, so only C is dependent on AB and AB can not define every attribute in the relation so AB cannot be a candidate key.

AE can be the candidate key of this relation, because from A we can define B and from AB we can define C, from CE we can define D, so from AE we can make sure all attributes are distinct so AE can be a candidate key.

Question 3. Minimal Cover of a set of Functional Dependencies [20 marks]
Consider the set of functional dependencies $F = \{A \rightarrow B, B \rightarrow CD, D \rightarrow A, AC \rightarrow D\}$. Compute a minimal cover of F. Justify your answer.

- Apply the Decomposition Inference Rule

G1={A->B,B->C,B->D,D->A,AC->D}

- Do left reduction

To test whether there is a superfluous attribute on the LHS, we try to remove each of the LHS attributes and apply attribute closure algorithm to see if the RHS still functionally depends on the remainder of the LHS:

(AC-C)+= A+ = ABCD ,because (AC - C) + contains D so A->D

(AC-A)+= C+ = C, (AC - A) + does not contains D

So after left reduction :

G2={A->B,B->C,B->D,D->A,A->D}

- Eliminate Redundant FDs

A+=ABCD

B+=BCDA

C+=C

D+=D

ABC

G3={

Question 4. 3NF Normalization [25 marks]

Consider a relation schema $N(R, F)$ where $R = \{A, B, C, D\}$ and $F = \{A \rightarrow B,\ C \rightarrow D\}$. Perform the following tasks. Justify your answers.

1) [5 marks] Identify all keys for *N*. Show your process.

A+=AB

B+=B

C+=CD

D+=D

AB+=AB

AC+=ABCD

AD+=ABD

BC+=BCD

BD+=BD

CD+=CD

ABC+=ABCD

ABD+=ABD

BCD+=BCD

ACD+=ABCD

from above we can see that the key can be AC,ABC and ACD as they could cover all attributes and the key should be AC because it has minimal attributes.

2) [5 marks] Identify the highest normal form (1NF, 2NF, 3NF, BCNF) that *N* satisfies.

1NF , because A->B and C->D are all partially dependent on the key AC so it is 1NF.

3) [10 marks] If *N* is not in 3NF, compute a lossless transformation into a set of 3NF relation schemas using the Synthesis algorithm.

- Find a minimal cover:

  already have it : $\{A \rightarrow B,\ C \rightarrow D\}$

- group FDs according to LHS and make a schema for each group

  (*A*->*B*)(*C*->*D*)

  S = {({A, B }, {A }), ({C, D }, {D })}

- If none of relation schemes in S contain a key of (U, F), create a new relation scheme in S that will contain only a key of (U, F)

  S = {({A, B }, {A }), ({C, D }, {C }),({A,C},{AC})}

4) [5 marks] Verify explicitly that your result has the lossless property, satisfies 3NF, and that all functional dependencies are preserved.

- Attribute preservation

  We can union all attributes : U= (A,B) U (C,D) U(A,C) = {A,B,C,D},which all the attributes are preserved and it also meets Lossless join decomposition.

- Functional dependency preservation

  From the union of dependency F, we can get A->B,C->D,A->C which all FDs are preserved.

- satisfies 3NF

  from 3) we get {({A, B }, {A }), ({C, D }, {C }),({A,C},{AC})}

  with dependency A->B,C->D,A->C for each relation

  for A->B, A is the super key of the relation{A,B}

  for C->D, C is the super key of the relation{C,D}

  for A->C, C is a prime attribute

Question 5. BCNF Normalization [30 marks]

Consider a relation schema N (R, F), where $R = \{A, B, C, D\}$ and $F = \{A \rightarrow C, D \rightarrow B, BC \rightarrow A, BC \rightarrow D\}$. Perform the following tasks. Justify your answers.

1) [5 marks] Identify all keys for N. Show process.

A+=AC

B+=B

C+=C

D+=DB

AB+=ABC

AC+=AC

AD+=ACDB

BC+=BCAD

BD+=BD

CD+=CDB

ABC+=ABC

ACD+=ACDB

BCD+=BCAD

from above we can see that the key can be AD,BC,ACD and BCD as they could cover all attributes and the key should be AD or BC because they have minimal attributes.

2) [4 marks] Identify the highest normal form (1NF, 2NF, 3NF, BCNF) that N satisfies.

For this question , I identify BC as the minimal key.

3NF because for A->C , C is a prime attribute and forD->B, B is a prime attribute, forBC->A and BC->D , BC is the super key so it is 3NF.

3) [16 marks] If *N* is not in BCNF, compute a lossless decomposition into a set of BCNF relation schemas using the BCNF decomposition algorithm.

R = {A, B, C, D}

F = {A→C, D→B, BC→A, BC→D}

first we can tell A->C violate BCNF

so

R = {A, B, D}

F = { D→B}

R2= {A C}

F2 = {A→C}

now the D->B still violate BCNF because the key for R1 is AD

so

R1= {A, C}

F1 = {A→C}

R = {A,D}

F = { null}

R2= {B,D}

F2={D->B}

Now R is BCNF but BC→A, BC→D is lost

so

R1= {A C}

F1 = {A→C}

R = {A,D}

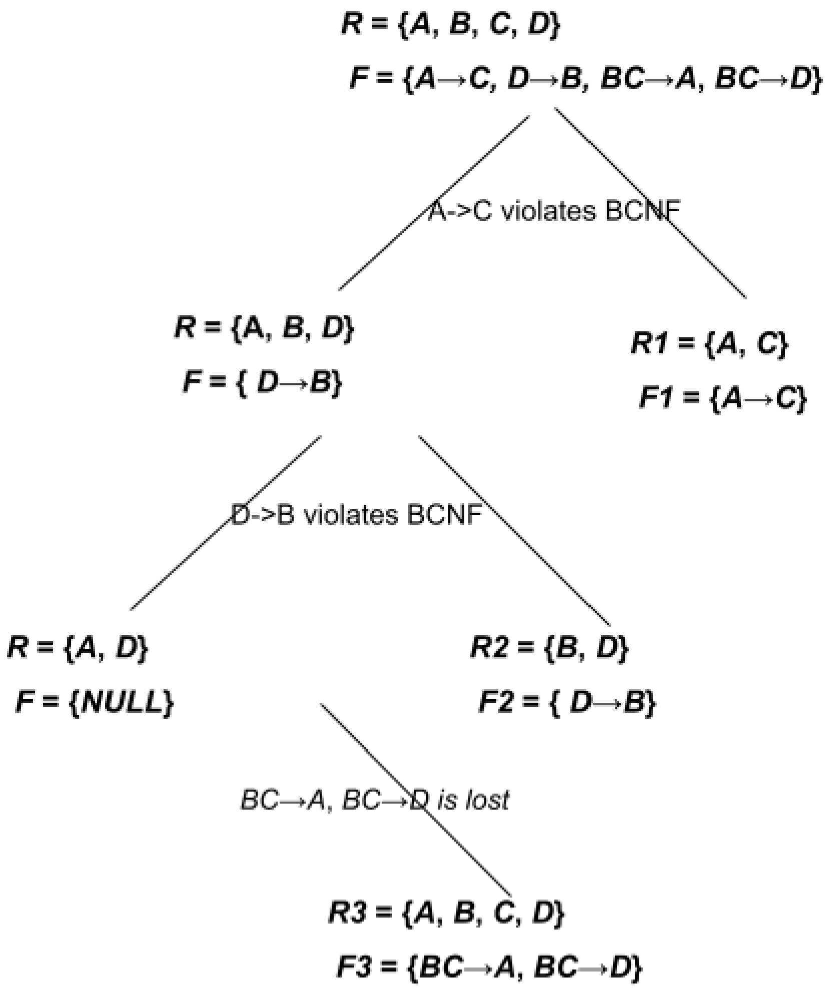
F = { null}

R2= {B,D}

F2={B->D}

R3={A, B, C, D}

F3={BC→A, BC→D}



Now they are all BCNF

4) [5 marks] Verify explicitly whether your result satisfies BCNF, and that all functional dependencies are preserved.

- Attribute preservation

  We can union all attributes : U= (A,B) U (B,D) U(A,C) U(A,B,C,D) = {A,B,C,D},which all the attributes are preserved and it also meets Lossless join decomposition.

- Functional dependency preservation

  From the union of dependency F, we can get A→C, D→B, BC→A, BC→D which all FDs are preserved.

- satisfies BCNF

  from 3) we get

  R1= {A C}

  F1 = {A→C}

  R = {A,D}

  F = { null}

  R2= {B,D}

  F2={B->D}

R3={A, B, C, D}

F3={BC→A, BC→D}

for R1= {A,C} F1 = {A→C}

A is the relation key of {A,C},so BCNF

for R2= {B,D} F2={B->D}

B is the relation key of{B,D},so BCNF

for R3={A, B, C, D} F3={BC→A, BC→D}

BC is the relation key of{A,B,C,D},so BCNF

****