Assignment 3 Part 2

Members:

- **Chris On (1003997071)**
- **Xinyue Jiang (1003830937)**

Part 2:

Question 1:

Given: Relation A with attributes LMNOPQRS with functional dependencies D:

D = { L → NQ, MNR → O, O → M, NQ → LS, S → OPR}

1. (a)

To find which functional dependencies violate BCNF, we will take the closure of the LHS for every functional dependency provided to check whether the it is a superkey. If the closure of the LHS of a FD does not functionally determine every attribute (a superkey) then the FD violates BCNF:

Taking the closures of the LHS of every FD we get:

a) L^+ = LNQSOPRM (is a superkey)
b) MNR^+ = MNRO (not a superkey)
c) O^+ = OM (not a superkey)

d) NQ^+ = NQLSPRMO (is a superkey)

e) S^+ = SOPRM (not a superkey)

Therefore, the following functional dependencies violate BCNF:

- MNR → O,
- O →M,
- S → OPR

1. (b)

We will decompose A with our choice functional dependency as:

- S → OPR

We will split A to A1 and A2 where:

- A1: LQNS,
- A2: MOPRS

Note: During our projections, we will stop projections early if an FD results in violating BCNF since we will be required to split the current relation projection. We also will not include following rows that include attributes that are superkeys or attributes that result in a trivial closure (ex. A → A).

- **A1: LQNS**

<table>
<tr><th>L</th><th>Q</th><th>N</th><th>S</th><th>closure</th><th>FDs?</th><th>Violate?</th></tr>
<tr><td>√</td><td></td><td></td><td></td><td>L^+= LNQSOPRM</td><td>L $\rightarrow$ NQS</td><td>No</td></tr>
<tr><td></td><td>√</td><td></td><td></td><td>Q^+ = Q</td><td>/</td><td>No</td></tr>
<tr><td></td><td></td><td>√</td><td></td><td>N^+ = N</td><td>/</td><td>No</td></tr>
<tr><td></td><td></td><td></td><td>√</td><td>S^+ = OPRM</td><td>/</td><td>No</td></tr>
<tr><td>√</td><td>√</td><td></td><td></td><td rowspan="3" colspan="3">irrelevant</td></tr>
<tr><td>√</td><td></td><td>√</td><td></td></tr>
<tr><td>√</td><td></td><td></td><td>√</td></tr>
<tr><td></td><td>√</td><td>√</td><td></td><td>QN^+=QNLSOPRM</td><td>QN $\rightarrow$ LS</td><td>No</td></tr>
<tr><td></td><td>√</td><td></td><td>√</td><td>QS^+=QSOPRM</td><td>/</td><td>No</td></tr>
<tr><td></td><td></td><td>√</td><td>√</td><td>NS^+=NSOPRM</td><td>/</td><td>No</td></tr>
<tr><td colspan="4">⋮</td><td colspan="3">irrelevant</td></tr>
</table>

- **A2: MOPRS**

M	O	P	R	S	Closure	FDs?	Violate?
√					$M^+ = M$	/	No
	√				$O^+ = OM$	O → M	Yes
		√			$P^+ = P$	/	No
			√		$R^+ = R$	/	No
				√	$S^+ = SOPRM$	S → MOPR	No
		⋮			We have already found a violation,		
		⋮			no need to check the rest		

Since we encountered a violation during our projections on A2, we will decompose A2 using the violating function dependency:

- O → M

We will split A2 to A3 and A4 where:

- A3: PRSO,
- A4: OM

- **A3: PRSO**

P	S	R	O	closure	FDs?	Violate?
√				$P^+ = P$	/	No
	√			$S^+ = SOPRM$	$S \rightarrow OPR$	No
		√		$R^+ = R$	/	No
			√	$O^+ = OM$	/	No
√	√			irrelevant		
√		√		$PR^+ = PR$	/	No
√			√	$PO^+ = POM$	/	No
	√	√		irrelevant		
	√		√			
		√	√	$RO^+ = ROM$	/	No
√		√	√	$PRO^+ = PROM$	/	No
⋮				irrelevant		

- **A4: OM**

O	M	closure	FDs?	Violate
√		$O^+ = OM$	$O \rightarrow M$	No
	√	$M^+ = M$	/	No

Final Decomposition of A:

- LNQS
- MO
- OPRS

Decomposition	FDs
LNQS	L → NQS, NQ → LS
MO	O → M
OPRS	S → OPR

Question 2

Given: Relation P with attributes ABCDEFGH with functional dependencies T:

T = { AB → C, C → ABD, CFD → E, E → B, BF → EC, B → DA }

2. (a)

We will compute the minimal basis for T using the follow process:

- Step 1: Split RHS of the functional dependencies
- Step 2: Try to reduce the LHS of the resulting functional dependencies

- Step 3: Eliminate any redundant functional dependencies by computing the cover of the LHS from the functional dependency with the approach shown in lecture (can only use FDs that are not removed and cannot use the current FD under consideration)

Step 1: split RHS

(a) AB → C

(b) C → A

(c) C → B

(d) C → D

(e) CFD → E

(f) E → B

(g) BF → C

(h) BF → E

(i) B → A

(j) B → D

Step 2: Reduce LHS

(a) B → C

(b) C → A

(c) C → B

(d) $C \rightarrow D$

(e) $CF \rightarrow E$

(f) $E \rightarrow B$

(g) $B \rightarrow C$

(h) $BF \rightarrow E$

(i) $B \rightarrow A$

(j) $B \rightarrow D$

Step 3: Eliminate FDs

(a) $B^{+}_{T-(a)} = BDA\underline{C}$ we can remove (a)

(b) $C^{+}_{T-(a)-(b)} = CBD\underline{A}$ we can remove (b)

(c) $C^{+}_{T-(a)-(b)-(c)} = CD$ **we cannot remove (c)**

(d) $C^{+}_{T-(a)-(b)-(d)} = CBA\underline{D}$ we can remove (d)

(e) $CF^{+}_{T-(a)-(b)-(d)-(e)} = CFBAD\underline{E}$ we can remove (e)

(f) $E^{+}_{T-(a)-(b)-(d)-(e)-(f)} = E$ **we cannot remove (f)**

(g) $B^{+}_{T-(a)-(b)-(d)-(e)-(g)} = BAD$ **we cannot remove (g)**

(h) $BF^{+}_{T-(a)-(b)-(d)-(e)\_(h)} = CBFAD$ **we cannot remove (h)**

(i) $B^{+}_{T-(a)-(b)-(d)-(e)-(i)} = BCD$ **we cannot remove (i)**

(j) $B^{+}_{T-(a)-(b)-(d)-(e)-(j)} = BCA$ **we cannot remove (j)**

Therefore, we result in the minimal basis for T as:

- B → A
- B → C
- B → D
- BF → E
- C → B
- E → B

2. (b)

We have the following cases:

- **Case 1 (LHS: X RHS: X) and Case 2 (LHS: √ RHS: X)**
  - These attributes cannot be retrieved from any functional dependencies therefore attributes in case 1 and 2 must be in the key.
- **Case 3 (LHS: X RHS: √)**
  - Is in no key, since these attributes do not functionally determine any other attributes and are themselves determined by another attribute through some functional dependency.
- **Case 4 (LHS: √ RHS: √)**
  - These are the attributes that we must check by computing the closure to determine whether closure under

consideration results in a being a key.

Note: Similar to before, we will not include following rows that include an attribute that resulted in a key from previous closure calculations.

- Organizing the attributes into their corresponding case:

Case	LHS	RHS	Attributes
1)	×	×	G, H
2)	√	×	F
3)	×	√	A, D
4)	√	√	B, C, E

- We will check the closures for each attribute in case 4.

B	C	E	Closure	Key?
√			$BFGH^+$ =BACDFGHE	YES
	√		$CFGH^+$ =CBADEFGH	YES
		√	$EFGH^+$ =EBFGHACD	YES
√	√		------------	--------

	√	√	------------	--------
√		√	------------	--------

The keys for P are:

- BFGH, CFGH, and EFGH.

2. (c)

From our computed minimal basis in **(a)**:

- B → A
- B → C
- B → D
- BF → E
- C → B
- E → B

From the 3NF synthesis algorithm, we would result in one relation per functional dependency; however, we can merge the RHS of our computed minimal basis which will yield a smaller set of relations that still form a lossless and dependency-preserving decomposition of P into a collection of relations that are 3NF

Merging our computed minimal basis results in:

- B → ACD
- BF → E
- C → B
- E → B

This will result in relations:

- P_1(A, B, C, D)
- P_2(B, E, F)
- P_3(B, C)
- P_4(B, E)

Since the attributes of BC are contained in P_1, we do not need the relation P_3, similarly since the attributes BE are contained in P_2 we do not need the relation P_4.

- Since relations P_1 and P_2 do not contain a key, we must add an additional relation P_5(B, F, G, H) where P_5 contains a key

Therefore, we result in a final set of relations:

- P_1(A, B, C, D)
- P_2(B, E, F)
- P_5(B, F, G, H)

2. (d)

Since we formed each relation from an FD, the LHS of those FDs are in fact superkeys for their relations; however, there may be other FDs that violate BCNF and therefore allow redundancy. We must find project the FDs onto each relation to check.

Without doing the full projection procedure, we can observe the functional dependency E → B. Clearly E → B will project onto the relation P_2(B, E, F) and the closure for this functional dependency results in:

- E+ = EBACD

Therefore, E is not a superkey of relation P_2(B, E, F) since it is missing F. So, we may conclude that our schema does indeed allow redundancy.