**CSC343 Assignment 3 Part 2**

**1.**

1. Follow the FDs of relation A, we can get:

L -> LMNOPQRS; L does not violate BCNF.

MNR -> MNRO; MNR is not a superkey, violation.

O -> OM; O is not a superkey, violation.

NQ -> LMNOPQRS; NQ does not violate BCNF.

S -> SOPRM; S is not a superkey, violation.

FD LHS & Superkey: L, NQ

Therefore, FDs in this relation violate BCNF are:

**MNR -> O; O -> M; S -> OPR**

1. Start with an FD that violates BCNF: O -> M;

Split relation A: A1: MO, A2: LNOPQRS,

Check A1 for BCNF violation:

A1 only has two attributes, no violation possible.

Check A2 for BCNF violation:

|  |  |
| --- | --- |
| **Closure** | **FDs** |
| L+= LMNOPQRS | Superkey, no violation |
| N+= N | Nothing |
| O+= MO | Nothing |
| P+= P | Nothing |
| Q+= Q | Nothing |
| R+= R | Nothing |
| S+= SOPRM | violates FD S -> OPR, abort projection |

Split A2: A3: OPRS, A4: LNQS

Check A3 for BCNF violation:

(we don’t check all supersets because most of them don’t even partially exist on the LHS of a none superkey, therefore no violation possible)

|  |  |
| --- | --- |
| **Closure** | **FDs** |
| O+= MO | Nothing |
| P+= P | Nothing |
| R+= R | Nothing |
| S+= SOPRM | Superkey of A3, no violation |

No violation found

Check A4 for BCNF violation:

(we don’t check all supersetS because of the same reason above)

|  |  |
| --- | --- |
| **Closure** | **FDs** |
| L+= LMNOPQRS | Superkey, no violation |
| N+= N | Nothing |
| Q+= Q | Nothing |
| S+= SOPRM | Nothing |
| LS+= LMNOPQRS | Any superset of L is a Superkey, no violation |
| NQ+= LMNOPQRS | Any superset of NQ is a Superkey, no violation |
| Any other superset | Cannot generate N/Q/S, no violation possible |

No violation found

Final decomposition (alphabetically):

**A4 = (LNQS) A1 = (MO) A3 = (OPRS)**

**L -> NQ, NQ -> LS O -> M S -> OPR**

**2.**

1. The basis for T, called P1:

1. AB -> C

2. B -> A

3. B -> D

4. BF -> C

5. BF -> E

6. C -> A

7. C -> B

8. C -> D

9. CFD -> E

10. E -> B

Eliminate redundant FDs

|  |  |  |  |
| --- | --- | --- | --- |
| **FD** | **Exclude when computing closures** | **Closure** | **Decision** |
| 1 | 1 | AB+= ABD, cannot get C | keep |
| 2 | 2 | B+= BD, cannot get A | keep |
| 3 | 3 | B+= BACD, can get D | discard |
| 4 | 3, 4 | BF+= BFACED, can get C | discard |
| 5 | 3, 4, 5 | BF+= BFACDE, can get E | discard |
| 6 | 3, 4, 5, 6 | C+= CBAD, can get A | discard |
| 7 | 3, 4, 5, 6, 7 | C+= CD, cannot get B | keep |
| 8 | 3, 4, 5, 6, 8 | C+= CBA, cannot get D | keep |
| 9 | 3, 4, 5, 6, 9 | CFD+= CFDBA, cannot get E | keep |
| 10 | 3, 4, 5, 6, 10 | E+= E, cannot get B | keep |

Remaining FDs, called P2:

1. AB -> C

2. B -> A

7. C -> B

8. C -> D

9. CFD -> E

10. E -> B

Reducing LHS:

1. AB -> C

B+= ABC… so reduce LHS to B.

9. CFD -> E

C+= CBD… so reduce LHS to CF.

The reduced set, called P3:

1. B -> C

2. B -> A

7. C -> B

8. C -> D

9. CF -> E

10. E -> B

Eliminate redundant FDs

|  |  |  |  |
| --- | --- | --- | --- |
| **FD** | **Exclude when computing closures** | **Closure** | **Decision** |
| 1 | 1 | B+= BA, cannot get C | keep |
| 2 | 2 | B+= BCD, cannot get A | keep |
| 7 | 7 | C+= CD, cannot get B | keep |
| 8 | 8 | C+= CBA, cannot get D | keep |
| 9 | 9 | CF+= CFDBA, cannot get E | keep |
| 10 | 10 | E+= E, cannot get B | keep |

No further simplifications are possible.

So the following set P4 is a minimal basis:

**1. B -> C**

**2. B -> A**

**7. C -> B**

**8. C -> D**

**9. CF -> E**

**10. E -> B**

1. From the minimal basis, we can get:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **LHS** | **RHS** | **Conclusion** |
| G, H | - | - | Must be in every key |
| E, F | Yes | - | Must be in every key |
| A, D | - | Yes | Is not in any key |
| B, C | Yes | Yes | Must check |

BEFGH+= BEFGHACD. So yes BEFGH is a key.

CEFGH+= CEFGHBAD. So yes CEFGH is also a key.

Therefore, all keys are as follows:

**BEFGH, CEFGH.**

1. Revised FDs from the minimal basis, called P5:

B -> AC

C -> BD

CF -> E

E -> B

After adding G & H that does not exist in any FDs, the set of relations that would result would have these attributes:

R1(A, B, C, G, H) R2(B, C, D, G, H) R3(C, E, F, G, H) R4(B, E, G, H)

Note, none of the relation attributes occur within another relation, so all relations are kept. Also, CEFGH is a key of P, there is no need to add a relation that includes a key.

The final set of relations is:

**R1(A, B, C, G, H) R2(B, C, D, G, H) R4(B, E, G, H)**

1. Because we formed each relation from an FD, the LHS of those FDs are indeed superkeys for their relations. However, there may be other FDs that violates BCNF and therefore allow redundancy. The only way to find out is to project the FDs onto each relation.

We can quite quickly find a relation that violates BCNF without doing all the full projections:

Clearly, E -> B will project onto the relation R4. And E+ = EB so E is not a superkey of this relation. So yes, **these schema allows redundancy.**