Solutions to Homework06

1. Given a relation schema ABCDEFG satisfying the following functional dependencies, find all keys.

   - $A \rightarrow I$
   - $AB \rightarrow C$
   - $AE \rightarrow GH$
   - $BE \rightarrow DF$
   - $H \rightarrow A$

   Solution:
   Attributes on the left: BE
   Attributes on the right: CDFG
   Attributes on both sides: AH
   Attributes never appears: N/A

   (a) Every key needs to include B and E. First We compute (BE)+ = BDEF

   (b) Compute (ABE)+ = ABCDEFGHI = R
   Then ABE is a key.

   (c) Compute (BEH)+ = ABCDEFGHI = R
   Then BEH is a key.

2. Given a relation schema ABCDEFGH, show that the given functional dependencies is a minimal cover.

   - $A \rightarrow B$
   - $ADE \rightarrow C$
   - $ADF \rightarrow G$
   - $CF \rightarrow GH$

   Solution:

   (a) 1. $A \rightarrow B$
       2. $DE \rightarrow C$ (Try LHS simplification and remove A)
       3. $ADF \rightarrow G$
       4. $CF \rightarrow GH$

       (a) could only be stronger, so we compute the closure of DE using the original set of FDs, getting DE. As C is not included, we proved non-equivalence. We cannot replace the original FDs with this set (a).

   (b) 5. $A \rightarrow B$
       6. $AE \rightarrow C$ (Try LHS simplification and removes D)

7. ADF $\rightarrow$ G
8. CF $\rightarrow$ GH

(b) could only be stronger, so we compute the closure of AE using the original set of FDs, getting AEB. As C is not included, we proved non-equivalence. We cannot replace the original FDs with this set (b).

(c) 9. A $\rightarrow$ B
10. AD $\rightarrow$ C (Try LHS simplification and removes E)
11. ADF $\rightarrow$ G
12. CF $\rightarrow$ GH

(c) could only be stronger, so we compute the closure of AD using the original set of FDs, getting ADB. As C is not included, we proved non-equivalence. We cannot replace the original FDs with this set (c).

(d) 13. A $\rightarrow$ B
14. ADE $\rightarrow$ C
15. DF $\rightarrow$ G (Try LHS simplification and removes A)
16. CF $\rightarrow$ GH

(d) could only be stronger, so we compute the closure of DF using the original set of FDs, getting DF. As G is not included, we proved non-equivalence. We cannot replace the original FDs with this set (d).

(e) 17. A $\rightarrow$ B
18. ADE $\rightarrow$ C
19. AF $\rightarrow$ G (Try LHS simplification and removes D)
20. CF $\rightarrow$ GH

(e) could only be stronger, so we compute the closure of AF using the original set of FDs, getting ABF. As G is not included, we proved non-equivalence. We cannot replace the original FDs with this set (e).

(f) 21. A $\rightarrow$ B
22. ADE $\rightarrow$ C
23. AD $\rightarrow$ G (Try LHS simplification and removes F)
24. CF $\rightarrow$ GH

(f) could only be stronger, so we compute the closure of AD using the original set of FDs, getting ABD. As G is not included, we proved non-equivalence. We cannot replace the original FDs with this set (f).

(g) 25. A $\rightarrow$ B

26. ADE $\rightarrow$ C
27. ADF $\rightarrow$ G
28. CF $\rightarrow$ H (Try RHS simplification and removes G)

(g) could only be weaker, so we compute the closure of CF using (g), getting CFH. As G is not included, we proved non-equivalence. We cannot replace the original FDs with this set (g).

(h)
29. A $\rightarrow$ B
30. ADE $\rightarrow$ C
31. ADF $\rightarrow$ G
32. CF $\rightarrow$ G (Try RHS simplification and removes H)

(h) could only be weaker, so we compute the closure of CF using (h), getting CFG. As H is not included, we proved non-equivalence. We cannot replace the original FDs with this set (h).

(i)
33. A $\rightarrow$ B
34. ADE $\rightarrow$ C
35. ADF $\rightarrow$ G
36. F $\rightarrow$ GH (Try LHS simplification and removes C)

(i) could only be stronger, so we compute the closure of F using the original set of FDs, getting F. As GH is not included, we proved non-equivalence. We cannot replace the original FDs with this set (i).

(j)
37. A $\rightarrow$ B
38. ADE $\rightarrow$ C
39. ADF $\rightarrow$ G
40. C $\rightarrow$ GH (Try LHS simplification and removes F)

(j) could only be stronger, so we compute the closure of C using the original set of FDs, getting C. As GH is not included, we proved non-equivalence. We cannot replace the original FDs with this set (j).

Therefore, no simplification can be made and the given set of functional dependencies is a minimal cover.

3. Given a relation schema ABCDEFGH satisfying the following functional dependencies, find a minimal cover.
   - A $\rightarrow$ BC
   - AB $\rightarrow$ D
   - B $\rightarrow$ C

Solution:

(a)
- A → B (Try RHS simplification and removes C)
- AB → D
- B → C

(a) could only be weaker, so we compute the closure of A using (a), getting ABCD. As C is included, we proved equivalence. We can replace the original set of FDs with (a).

(b)
- A → B
- A → D (Try LHS simplification and removes B)
- B → C

(b) could only be stronger. We compute the closure of A using (a), getting ABDC. As D is included, we proved equivalence. We can replace (a) with (b).

(c)
1. A → BD (Union Rule)
2. B → C

No further LHS simplification or RHS simplification can be done. We obtain the minimal cover.
A → BD
B → C

4. Given a relation schema ABCDEFGH satisfying the following functional dependencies, find a minimal cover.
   - A → HI
   - AB → CD
   - CD → EF
   - E → F
   - G → AD
   - H → B
   - I → AG

   Solution:

   (a) There is no union rule can be applied for now.

   (b)
   1. A → I (Try RHS simplification and removes H)
   2. AB → CD
   3. CD → EF
   4. E → F

5. $G \rightarrow AD$
6. $H \rightarrow B$
7. $I \rightarrow AG$

(b) could only be weaker. We compute the closure of A using (b), getting AIGD. As H is not included, we proved non-equivalence. We cannot replace the original FDs with (b).

(c) 8. $A \rightarrow H$ (Try RHS simplification and removes I)
9. $AB \rightarrow CD$
10. $CD \rightarrow EF$
11. $E \rightarrow F$
12. $G \rightarrow AD$
13. $H \rightarrow B$
14. $I \rightarrow AG$

(c) could only be weaker, so we compute the closure of A using (c), getting AHBCDEF . As I is not included, we proved non-equivalence. We cannot replace the original FDs with (c).

(d) 15. $A \rightarrow HI$
16. $AB \rightarrow C$ (Try RHS simplification and removes D)
17. $CD \rightarrow EF$
18. $E \rightarrow F$
19. $G \rightarrow AD$
20. $H \rightarrow B$
21. $I \rightarrow AG$

(d) could only be weaker, so we compute the closure of AB using (d), getting ABCHIBGDEF . As D is included, we proved equivalence. We can replace the original FDs with (d).

(e) 22. $A \rightarrow HI$
23. $A \rightarrow C$ (Try LHS simplification and removes B)
24. $CD \rightarrow EF$
25. $E \rightarrow F$
26. $G \rightarrow AD$
27. $H \rightarrow B$

28. I $\rightarrow$ AG

(e) could only be stronger, so we compute the closure of A using (d), getting AHIBCGDEF . As B is included, we proved equivalence. We can replace (d) with (e).

(f) 29. A $\rightarrow$ CHI (union rule)

30. CD $\rightarrow$ EF

31. E $\rightarrow$ F

32. G $\rightarrow$ AD

33. H $\rightarrow$ B

34. I $\rightarrow$ AG

(f) union rule

(g) 35. A $\rightarrow$ CHI

36. CD $\rightarrow$ E (Try RHS simplification and removes F)

37. E $\rightarrow$ F

38. G $\rightarrow$ AD

39. H $\rightarrow$ B

40. I $\rightarrow$ AG

(g) could only be weaker. We compute the closure of CD using the (g), getting CDEF. As F is included, we proved equivalence. We can replace the (f) with (g).

(h) 41. A $\rightarrow$ CHI

42. C $\rightarrow$ E (Try LHS simplification and removes D)

43. E $\rightarrow$ F

44. G $\rightarrow$ AD

45. H $\rightarrow$ B

46. I $\rightarrow$ AG

(h) could only be stronger, so we compute the closure of C using (g), getting C . As E is not included, we proved non-equivalence. We cannot replace (g) with (h). Same for removing C in CD$\rightarrow$ E.

(i) 47. A $\rightarrow$ CHI

48. CD $\rightarrow$ E

49. E $\rightarrow$ F

50. G $\rightarrow$ D (Try RHS simplification and removes A)

51. H $\rightarrow$ B

52. I $\rightarrow$ AG

(i) could only be weaker. We compute the closure of G using the (i), getting GD. As A is not included, we proved non-equivalence. We cannot replace the (h) with (i).

(j)
53. A $\rightarrow$ CHI
54. CD $\rightarrow$ E
55. E $\rightarrow$ F
56. G $\rightarrow$ A (Try RHS simplification and removes D)
57. H $\rightarrow$ B
58. I $\rightarrow$ AG

(j) could only be weaker. We compute the closure of G using the (j), getting GACHIB. As D is not included, we proved non-equivalence. We cannot replace the (h) with (j).

(k)
59. A $\rightarrow$ CHI
60. CD $\rightarrow$ E
61. E $\rightarrow$ F
62. G $\rightarrow$ AD
63. H $\rightarrow$ B
64. I $\rightarrow$ G (Try RHS simplification and removes A)

(k) could only be weaker. We compute the closure of I using the (k), getting IGADCHEFB. As A is included, we proved equivalence. We can replace the (h) with (k).

No further LHS simplification or RHS simplification can be done. We obtain the minimal cover.
A $\rightarrow$ CHI
CD $\rightarrow$ E
E $\rightarrow$ F
G $\rightarrow$ AD
H $\rightarrow$ B
I $\rightarrow$ G

5. Given the following minimal cover, create a 3NF decomposition.

- ABD $\rightarrow$ G
- AG $\rightarrow$ E
- BD $\rightarrow$ C
- CF $\rightarrow$ A
- G $\rightarrow$ B

Solution:

(a) Creating tables from this minimal cover:
We get:

1. Table ABDG (ABD key)
2. Table AGE (AG key)
3. Table BDC (BD key)
4. Table CFA (CF key)
5. Table GB (G key)

(b) Check redundancy of tables and there is a redundant table of GB because table ABDG already contains GB. So, remove Table GB.

(c) Computing global key:
Attributes on the left: DF
Attributes on the right: E
Attributes on both sides: ABCG
Attributes never appears: N/A
Start from (DF)+ = DF
(ADF)+ = ADF
(BDF)+ = ABCDEFG = R
Therefore, BDF is the global key and we need to create a table to store BDF. GDF is also a global key, so table GDF is acceptable as well.

(d) The 3NF decomposition is:

1. Table ABDG (ABD key)
2. Table AGE (AG key)
3. Table BDC (BD key)
4. Table CFA (CF key)
5. Table BDF (BDF key)

(e) If with the original version of question 5, Given relation schema R = ABCDEFGH
Computing global key:
Attributes on the left: DF
Attributes on the right: E
Attributes on both sides: ABCG
Attributes never appears: H
Start from (DFH)+ = DFH
(ADFH)+ = ADFH
(BDFH)+ = ABCDEFGH = R
Therefore, BDFH is a global key and we need to create a table to store BDFH. GDFH is a global key as well. Table GDFH is acceptable as well.

(f) If with the original version of question 5 with relation schema ABCDEFGH
The 3NF decomposition is:

1. Table ABDG (ABD key)
2. Table AGE (AG key)
3. Table BDC (BD key)
4. Table CFA (CF key)
5. Table BDFH (BDFH key)

6. Given a relation schema ABCDEFG and the following minimal cover, create a 3NF decomposition.

   - A $\rightarrow$ B
   - B $\rightarrow$ DE
   - CF $\rightarrow$ DE
   - DG $\rightarrow$ CF

   Solution:

   (a) Creating tables from this minimal cover:
   We get:

   1. Table AB (A key)
   2. Table BDE (B key)
   3. Table CFDE (CF key)
   4. Table DGCF (DG key)

   (b) There is no redundant table.

   (c) Check if there is at least one table that stores global key.

   (d) Computing key:
   Attributes on the left: AG
   Attributes on the right: E
   Attributes on both sides: BCDF
   Attributes never appears: N/A
   Start from (AG)+ = ABCDEFG = R
   Therefore, AG is the global key and we need to create a table to store AG.

   (e) The 3NF decomposition is:

   1. Table AB (A key)
   2. Table BDE (B key)
   3. Table CFDE (CF key)
   4. Table DGCF (DG key)
   5. Table AG (AG key)

(f) If with the original version of question 6: Given a relation schema ABCDEFGH
The first two steps will be the same until we computing the key.
Computing key:
Attributes on the left: AG
Attributes on the right: E
Attributes on both sides: BCDF
Attributes never appears: H
Start from (AGH)+ = ABCDEFGH = R
Therefore, AGH is the global key and we need to create a table to store AGH.

(g) The original version of question 6 with relation schema ABCDEFGH
The 3NF decomposition is:

1. Table AB (A key)
2. Table BDE (B key)
3. Table CFDE (CF key)
4. Table DGCF (DG key)
5. Table AGH (AGH key)