

1. 等价, 理由如下:

- 首先往证, $F^+ \subseteq G^+$.
 - 由于 $A \rightarrow CD \in G$, 所以有 $A \rightarrow C, A \rightarrow D$, 进而有 $AC \rightarrow D$, 从而 G 蕴含 $A \rightarrow C, AC \rightarrow D$;
 - 又由于 $E \rightarrow AH$, 有 $E \rightarrow A, E \rightarrow H$, 又 $A \rightarrow D \in G^+$, 从而有 $E \rightarrow D$, 故 G 蕴含 $E \rightarrow AD, E \rightarrow H$.

综上两点, 有 $F^+ \subseteq G^+$.

- 再证, $G^+ \subseteq F^+$.
 - 由于 $A \rightarrow C, A \rightarrow A$ 且有 $AC \rightarrow D$, 从而有 $AC \rightarrow D$, 进而有 $A \rightarrow CD$, 即 F 蕴含 $A \rightarrow CD$.
 - 又由于 $E \rightarrow AD$, 易有 $E \rightarrow A$, 又 $E \rightarrow H$, 从而有 $E \rightarrow AH$, 即 F 蕴含 $E \rightarrow AH$.

综上两点, 有 $G^+ \subseteq F^+$.

综上所述, 可以有 $F^+ = G^+$, 从而 F 和 G 等价.

2. (1) **3NF**. 由于候选键是 XY 和 XZ , 从而对于 X, Y 和 Z 而言均为键属性, 不存在非键属性对候选键的传递依赖和部分函数依赖, 所以至少为 3NF; 又由于键属性 Z 对候选键 XZ 存在部分函数依赖, 故其不是 BCNF. 所以其为 3NF.(2) **BCNF**. R 的候选键为 X 和 Y , 根据定义在函数依赖的左部均为候选键, 故其为 BCNF.(3) **1NF**. R 的候选键为 WX , 又由于 $X \rightarrow Z, WX \rightarrow X$, 所以存在 $WX \rightarrow Z$ 这个部分函数依赖, 所以其为 1NF.3. (1) 首先将函数依赖集 F 转化为对应的函数右部仅有 1 个属性的形式.

即 $F = \{E \rightarrow G, G \rightarrow E, F \rightarrow E, F \rightarrow G, H \rightarrow E, H \rightarrow G, FH \rightarrow E\}$.

(2) 去掉左部冗余的属性, 由于 $FH \rightarrow E, F \rightarrow E$, 从而 $FH \rightarrow E$ 冗余化简后为 $F = \{E \rightarrow G, G \rightarrow E, F \rightarrow E, F \rightarrow G, H \rightarrow E, H \rightarrow G\}$.(3) 去掉冗余的函数依赖, 由于 $F \rightarrow E, E \rightarrow G$ 可以得到 $F \rightarrow G$, 从而后者冗余; 同理可以有 $H \rightarrow G$ 冗余,

化简后的结果 $F = \{E \rightarrow G, G \rightarrow E, F \rightarrow E, H \rightarrow E\}$.

故最终的结果为 $F = \{E \rightarrow G, G \rightarrow E, F \rightarrow E, H \rightarrow E\}$.

4. 首先求得 $F_m = \{E \rightarrow D, BC \rightarrow D, CD \rightarrow A\}$, 进而求得保持函数依赖的关系模式分解为 $\rho = \{ED, BCD, ACD\}$ 5. (1) 根据 L 类的定义, B, E 是 L 类属性, 且此处不存在 N 类属性. 又 $X_{BE}^+ = \{ABCDE\}$, 从而 $\{BE\}$ 是唯一候选键.(2) 初始化后的结果如表 1 所示, 最终结果如表 6 所示, 可以看到没有任意一行中包含 a_1, a_2, a_3, a_4, a_5 , 故此处不是无损连接分解.

	A	B	C	D	E
AD	a_1	b_{12}	b_{13}	a_4	a_5
AB	a_1	a_2	b_{23}	b_{24}	b_{25}
BC	b_{31}	a_2	a_3	b_{34}	b_{35}
CDE	b_{41}	b_{42}	a_3	a_4	a_5
AE	a_1	b_{52}	b_{53}	b_{54}	a_5

图 1: 初始化

	A	B	C	D	E
AD	a_1	b_{12}	b_{13}	a_4	a_5
AB	a_1	a_2	b_{13}	b_{24}	b_{25}
BC	b_{31}	a_2	a_3	b_{34}	b_{35}
CDE	b_{41}	b_{42}	a_3	a_4	a_5
AE	a_1	b_{52}	b_{13}	b_{54}	a_5

图 2: $A \rightarrow C$

	A	B	C	D	E
AD	a_1	b_{12}	b_{13}	a_4	a_5
AB	a_1	a_2	b_{13}	b_{24}	b_{25}
BC	b_{31}	a_2	a_3	a_4	b_{35}
CDE	b_{41}	b_{42}	a_3	a_4	a_5
AE	a_1	b_{52}	b_{13}	b_{54}	a_5

图 3: $C \rightarrow D$

	A	B	C	D	E
AD	a_1	b_{12}	b_{13}	a_4	a_5
AB	a_1	a_2	a_3	b_{24}	b_{25}
BC	b_{31}	a_2	a_3	a_4	b_{35}
CDE	b_{41}	b_{42}	a_3	a_4	a_5
AE	a_1	b_{52}	b_{13}	b_{54}	a_5

图 4: $B \rightarrow C$

	A	B	C	D	E
AD	a_1	b_{12}	b_{13}	a_4	a_5
AB	a_1	a_2	a_3	b_{24}	b_{25}
BC	b_{31}	a_2	a_3	a_4	b_{35}
CDE	b_{41}	b_{12}	a_3	a_4	a_5
AE	a_1	b_{52}	b_{13}	b_{54}	a_5

图 5: $DE \rightarrow C$

	A	B	C	D	E
AD	a_1	b_{12}	b_{13}	a_4	a_5
AB	a_1	a_2	a_3	b_{24}	b_{25}
BC	b_{31}	a_2	a_3	a_4	b_{35}
CDE	b_{41}	b_{12}	a_3	a_4	a_5
AE	a_1	b_{52}	b_{13}	b_{54}	a_5

图 6: $CE \rightarrow A$

(3) 由于 BE 是键, 且不存在任意一个依赖的左部是 BE , 随机选择一个函数依赖即可开始算法.

- i. 对于 $A \rightarrow C$, A 不是候选键, 所以我们可以拆分成两个关系 $(AC), (ABDE)$. 则对于 $\langle AC, A \rightarrow C \rangle$, A 是主键, 从而 (AC) 是 BCNF; 又 $\langle ABDE, \emptyset \rangle$ 所有属性均为键属性, 从而其同样为 BCNF. 算法结束.

最终拆分的结果为 $(AC), (ABDE)$.

6. (1) 因为 $\{\text{课程编号}, \text{章节编号}, \text{学期}, \text{年}\}^+ = U$ 且 $\{\text{上课时间}, \text{教室}, \text{课程编号}, \text{章节编号}\}^+ = U$, 故候选键为 $\{\text{课程编号}, \text{章节编号}, \text{学期}, \text{年}\}$ 和 $\{\text{上课时间}, \text{教室}, \text{课程编号}, \text{章节编号}\}$, 并选择 **$\{\text{上课时间}, \text{教室}, \text{课程编号}, \text{章节编号}\}$ 为主键.**

(2) 由于 $\{\text{课程编号}\} \rightarrow \{\text{学院}, \text{课时}, \text{等级}\}$ 的左部不为候选键, 所以 R 不是 BCNF. 因此使用算法将其分解为 BCNF.

- i. 对于 $\{\text{课程编号}\} \rightarrow \{\text{学院}, \text{课时}, \text{等级}\}$, 其左部不为候选键, 故将 R 拆分为 2 个关系 $\{\text{课程编号}, \text{章节编号}, \text{教师编号}, \text{学期}, \text{年}, \text{上课时间}, \text{教室}, \text{学生数量}\}$ 和 $\{\text{学院}, \text{课时}, \text{等级}\}$. 并且对于两个关系, 其函数依赖的左部均为候选码 (第 2 个关系的函数依赖集为空集), 算法结束.

规范化后的结果为 {课程编号, 章节编号, 教师编号, 学期, 年, 上课时间, 教室, 学生数量} 和 {学院, 课时, 等级}