数据库作业week7

7.1

> Suppose that we decompose the schema $R = (A, B, C, D, E)$ into
>
> $$(A, B, C)$$
> $$(A, D, E)$$
>
> Show that this decomposition is a lossless decomposition if the following set F of functional dependencies holds:
>
> $$A \to BC$$
> $$CD \to E$$
> $$B \to D$$
> $$E \to A$$

because of $R_1 = (A, B, C)$,$R_2 = (A, D, E)$,$R_1 \cap R_2 = A$,so it is important to check if A is the superkey.

Now we compute Attribude of A. since of `result`:=A -> `result`:=ABC -> `result`:=ABCD -> `result`:=ABCDE .

so A is the superkey of R(same as R1\R2),this decomposition is a lossless decomposition.

7.13

> Show that the decomposition in Exercise 7.1 is not a dependency-preserving decomposition

because in the $CD \to E$,CDE are not include in any schema

Now we doing algorithm

$$for\ R_1$$
$$CD \to E$$
$$result := CD$$
$$result \cup ((result \cap R_1)_F^+ \cap R_1) = \{C, D\}$$
$$for R_2$$
$$result := CD$$
$$E \notin resulit = result \cup ((result \cap R_2)_F^+ \cap R_1) = \{C, D\}$$

beacuse $CD \to E$ is wrong in both R_1、R_2,so the decomposition in Exercise 7.1 is not a dependency-preserving decomposition

7.21

> Give a lossless decomposition into BCNF of schema *R* of Exercise 7.1.
>
> > 7.1 Suppose that we decompose the schema $R = (A, B, C, D, E)$ into
> >
> > $$(A, B, C)$$
> > $$(A, D, E)$$
> >
> > Show that this decomposition is a lossless decomposition if the following set F of functional dependencies holds:

> > $$A \to BC$$
> > $$CD \to E$$
> > $$B \to D$$
> > $$E \to A$$

Now we can know E is superkey

we should get the non-trivial dependency which left value is not E

$$for\ CD \to E$$
$$CD \cup E = \{C, D, E\}$$
$$R - E = \{A, B, C, D\}$$
$$\{C, D, E\}chooseCD \to E\ is\ BCNF$$
$$\{A, B, C, D\}chose\ A \to BC、B \to D、CD \to A$$
$$is\ not\ BCNF\ and\ A/CD\ is\ superkey$$
$$for\ B \to D$$
$$B \cup D = \{B, D\}$$
$$\{A, B, C, D\} - D = \{A, B, C\}$$
$$\{B, D\}choose\ B \to D\ is\ BCNF$$
$$\{A, B, C\}chose\ A \to BC\ is\ BCNF$$
$$so\ we\ get:$$
$$\{C, D, E\}choose\ CD \to E$$
$$\{B, D\}choose\ B \to D$$
$$\{A, B, C\}choose\ A \to BC$$

7.22

> Give a lossless, dependency-preserving decomposition into 3NF of schema *R* of Exercise 7.1.
>
> > 7.1 Suppose that we decompose the schema $R = (A, B, C, D, E)$ into
> >
> > $$(A, B, C)$$
> > $$(A, D, E)$$
> >
> > Show that this decomposition is a lossless decomposition if the following set F of functional dependencies holds:
> >
> > $$A \to BC$$
> > $$CD \to E$$
> > $$B \to D$$
> > $$E \to A$$

A\E is candidate key

$$after\ algorithm,\ F_c = F\ = \{A \to BC, CD \to E, B \to D, E \to A\}$$
$$candidate\ key\ are A, E$$
$$for\ A \to BC$$
$$R_1 = (A, B, C)\ choose A \to BC$$
$$for\ CD \to E$$
$$R_2 = (C, D, E) choose CD \to E$$
$$for\ B \to D$$
$$R_3 = (B, D) choose B \to D$$
$$for\ E \to A$$
$$R_4 = (E, A) choose E \to A$$
$$and\ we\ can\ verify\ the\ lossless\ of\ this\ decomposition\ because\ of\ R_4\ include\ AE$$

7.30

> Consider the following set F of functional dependencies on the relation schema (A, B, C, D, E, G):
>
> $$A \to BCD$$
> $$BC \to DE$$
> $$B \to D$$
> $$D \to A$$
>
> a. Compute B^+.

$$B^+ = B$$
$$\because B \to D$$
$$\therefore B^+ = BD$$
$$\because D \to A$$
$$\therefore B^+ = ABD$$
$$\because A \to BCD$$
$$\therefore B^+ = ABCD$$
$$\because BC \to DE$$
$$\therefore B^+ = ABCDE$$

> b. Prove (using Armstrong's axioms) that AG is a superkey.

$$AG \to A(reflexivity)$$
$$\because AG \to A, A \to BCD$$
$$\therefore AG \to BCD(transitivity)$$
$$\because AG \to BCD \to BC(reflexivity), BC \to DE$$
$$\therefore AG \to DE(transitivity)$$
$$\because AG \to BCD, AG \to DE$$
$$\therefore AG = AGAG \to AGBCD, AGBCD \to BCDE$$
$$\therefore AG \to BCDE$$
$$the\ same\ as\ AG \to BCDE, AG \to AG$$
$$\therefore AG \to ABCDEG$$
$$So\ AG\ is\ a\ superkey$$

> c. Compute a canonical cover for this set of functional dependencies F; give each step of your derivation with an explanation.

$$//canonical\ cover\ F_c\ = F$$
$$1.there\ are\ not\ dependancy\ which\ left\ arrtitudes\ are\ same\ as\ another\ dependency$$
$$2\ for\ A \to BCD, now\ check\ D, D \in A^+_{F'_c} = ABCDE, so\ delete\ D$$
$$now\ F_c = \{A \to BC, BC \to DE, B \to D, D \to A\}$$
$$1.there\ are\ not\ dependancy\ which\ left\ arrtitudes\ are\ same\ as\ another\ dependency$$
$$2.\ for\ BC \to DE, now\ check\ D, D \in BC^+_{F'_c} = ABCDE, so\ delete\ D$$
$$now\ F_c = \{A \to BC, BC \to E, B \to D, D \to A\}$$
$$1.there\ are\ not\ dependancy\ which\ left\ arrtitudes\ are\ same\ as\ another\ dependency$$
$$2.\ for\ BC \to D, now\ check\ C, C \in B^+_{F_c} = ABCDE, so\ delete\ C$$
$$now\ F_c = \{A \to BC, B \to E, B \to D, D \to A\}$$
$$1.union\ B \to E, B \to D\ toB \to DE$$
$$now\ F_c = \{A \to BC, B \to DE, D \to A\}$$
$$1.there\ are\ not\ dependancy\ which\ left\ arrtitudes\ are\ same\ as\ another\ dependency$$
$$2.\ for\ all\ dependency\ , there\ is\ no\ extraneous\ attribute$$
$$So\ canonical\ cover\ F_c = \{A \to BC, B \to DE, D \to A\}$$

> d. Give a 3NF decomposition of the given schema based on a canonical cover.

$$F_c = \{A \to BC, B \to DE, D \to A\}$$
$$candidate\ key\ are\ AG/BG/DG$$
$$for\ A \to BC$$
$$R_1 = (A, B, C)\ chooseA \to BC$$
$$for\ B \to DE$$
$$R_2 = (B, D, E) chooseB \to DE$$
$$for\ D \to A$$
$$R_3 = (A, D) chooseD \to A$$
$$for\ lossless\ requirement$$
$$R_4 = (A, G) choose\ \phi$$
$$and\ we\ can\ verify\ the\ lossless\ of\ this\ decomposition\ because\ of\ R_4\ include\ AG$$

> e. Give a BCNF decomposition of the given schema using the original set F of functional dependencies.

$$//BCNF, the\ superkey\ is A/B/D$$
$$for\ BC \rightarrow DE$$
$$BC \cup DE = \{B, C, D, E\} choose\ BC \rightarrow DE\ is\ BCNF$$
$$r - \{D, E\} = \{A, B, C, G\} choose B \rightarrow A, A \rightarrow BC\ is BCNF$$
$$So\ we\ get$$
$$\{B, C, D, E\} choose\ BC \rightarrow DE$$
$$\{A, B, C, G\} choose B \rightarrow A, A \rightarrow BC$$