COMP9311 Assignment 2 Haojin, Guo z5216214 17, Nov, 2020

**Question 1,**

**(1)**

+  iff +

Thus, need to compute {A}+

1st scan of A:

A+ := {A}

A+ := {A, B, C}

2nd scan of A:

A+ := {A, B, C}no change, therefore the algorithm terminates.

A+ := {A, B, C} , but +

Therefore, +

**(2)**

X = {A, C, D, E, H,G} can be considered as a super key,

because {A, C, D, E, H, G}+ = {A, B, C, D, E, G, H, I, J} but the attribute G is not in FD set.

Then, find a candidate key,

A can be removed because {C, D, E, H, G}+ = {A, B, C, D, E, G, H, I, J} = F,

(from CD 🡪AE) E also can be removed because {C, D, H, G}+ = {A, B, C, D, E, G, H, I, J} = F,

(from CD 🡪AE)

H can be further removed because {C, D, G}+ = {A, B, C, D, E, G, H, I, J} = F,

(from CD 🡪AE, E🡪CHI, H🡪J)

C can not be removed because {D, G}+ = {D, G}F

Similarly D,G can not be removed.

Therefore, CDG is a candidate key. (And the another candidate key is EDG.)

**(3)**

The highest normal form is 1NF.

1. In the FD set, for example, CD 🡪AE, E🡪CHI, H🡪J, it is found that attributes E and H are transmit module, so it’s transitive functional dependency, not 3NF.
2. When CDG chosen as the candidate key, CDG 🡪AE can be written as CD 🡪AE, so there is a partial dependency , not 2NF.
3. R(U) satisfy atomicity

Therefore, the highest normal form of R with respect to F is 1NF.

**(4)**

Fm = {A🡪B, A🡪C, CD🡪A, CD🡪E, E🡪C, E🡪H, E🡪I, H🡪J}

Step1: Fm = F

Step2: Replace A🡪BC with A🡪B, A🡪C.

Replace CD🡪AE with CD🡪A, CD🡪E.

Replace E🡪CHI with E🡪C, E🡪H, E🡪I.

Step3: C, D both can not be removed.

Step4: There are no redundant relations. No more FD could be removed, done.

**(5)**

The minimum cover is as above,

Fm = {A🡪B, A🡪C, CD🡪A, CD🡪E, E🡪C, E🡪H, E🡪I, H🡪J}

Meanwhile, Candidate key: (C, D, G)

Then, it can be decomposed as below.

R1’ = (A, B)

R2’ = (A, C)

R3’ = (C, D, A)

R4’ = (C, D, E)

R5’ = (E, C)

R6’ = (E, H)

R7’ = (E, I)

R8’ = (H, J)

None of relation contains a key of G, thus,

R9’ = (C, D, G)

Here, R1’ and R2’ can be merged as (A, B, C),

Similarly R3’ and R4’, R5’ , R6’ and R7’ , are also can be merged.

Therefore, the final decomposed relations are,

R1 = (A, B, C)

R2 = (C, D, A, E)

R3 = (E, C, H, I)

R4 = (H, J)

R5 = (C, D, G)

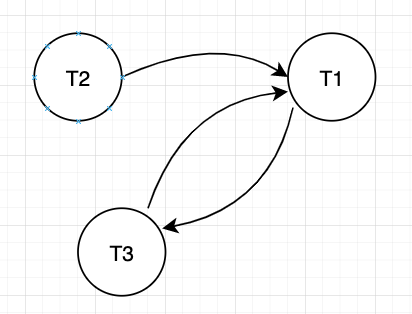
**Question 2,**

**(1)** Not serializable.

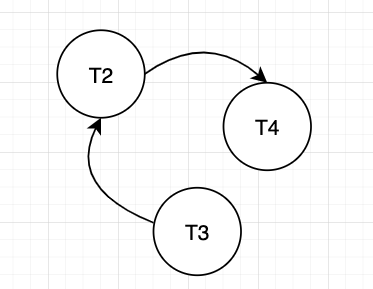
Precedence graph for this schedule:

Draw graphs separately based on the three values of A, B and C.

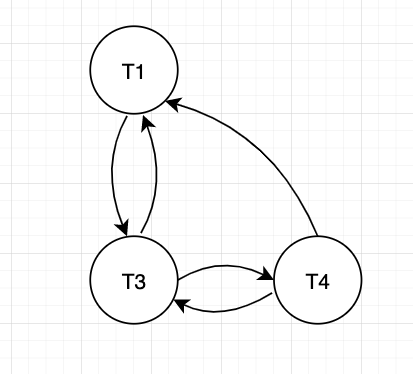
(i)For A:



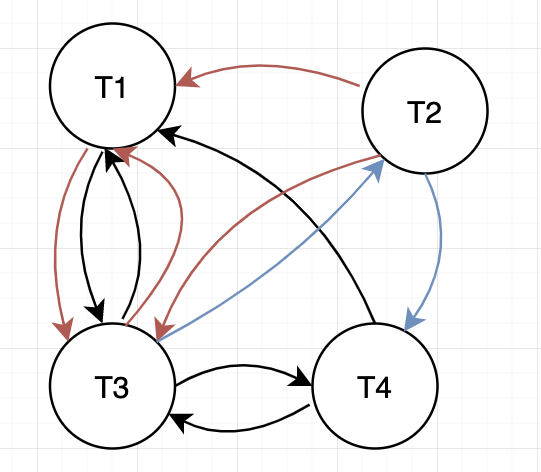
(ii)For B:



(iii)For C:



Put (i)(ii)(iii) together:



Has cycles according to the graphs above, therefore, this transaction schedule not serializable

(2)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Time | t1 | t2 | t3 | t4 | t5 | t6 | t7 | t8 | t9 | t10 | t11 | t12 | t13 | t14 | t15 | t16 | t17 | t18 |
| T1 | R(A) | R(C) | W(A) | W(C) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| T2 |  |  |  |  | R(A) | W(A) | R(B) | W(B) |  |  |  |  |  |  |  |  |  |  |
| T3 |  |  |  |  |  |  |  |  | R(B) | R(C) | R(A) | W(C) | W(B) | W(A) |  |  |  |  |
| T4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  | R(C) | W(C) | R(B) | W(B) |

(3)

|  |  |  |
| --- | --- | --- |
| Time | T1 | T2 |
| 1 |  | Write\_lock(A) |
| 2 |  | Read\_item(A) |
| 3 |  | Write\_item(A) |
| 4 |  | Unlock(A) |
| 5 | Write\_lock(A) |  |
| 6 | Read\_item(A) |  |
| 7 | Write\_lock(C) |  |
| 8 | Read\_item(C) |  |
| 9 | Write\_item(A) |  |
| 10 | Unlock(A) |  |
| 11 | Write\_item(C) |  |
| 12 | Unlock(C) |  |
| 13 |  | Write\_lock(B) |
| 14 |  | Read\_item(B) |
| 15 |  | Write\_item(B) |
| 16 |  | Unlock(B) |