**COMP9311 24T3: Assignment 2**

**Student ZID: z5518601**

**Name: Tianxiong Wu**

**Question 1 (12 marks)**

Consider a relation R (A, B, C, D, E, G, H, I, J) and its FD set 𝐹 = {𝐴𝐷 → 𝐵, 𝐵𝐷 →𝐺, 𝐵𝐸 → 𝐼, 𝐴𝐸 → 𝐷𝐼, 𝐴𝐼 → 𝐸, 𝐴𝐸𝐼 → 𝐶}.

Regarding the following questions, please give your answers and brief justifications.

1. **Check if 𝐴𝐵 → 𝐺. (1 mark)**

Knowing that, AD → B, BD → G since we can’t get AB → D then AB → G does not hold.

1. **Find all the candidate keys for R. (2 mark)**

As we know, 𝐴𝐷 → 𝐵, 𝐵𝐷 →𝐺, 𝐵𝐸 → 𝐼, 𝐴𝐸 → 𝐷𝐼, 𝐴𝐼 → 𝐸, 𝐴𝐸𝐼 → 𝐶

Let X: {A, E, J, H}

{A, E, J, H} + = {A, B, C, D, E, G, H, I, J}

Let X: {A, I, J, H}

{A, I, J, H} + = {A, B, C, D, E, G, H, I, J}

therefore, we can get AEJH, AIJH are the candidate keys for the relation R.

1. **Determine the highest normal form of R with respect to F. (2 marks)**

Since 𝐴𝐷 → 𝐵, 𝐵𝐷 →𝐺, 𝐵𝐸 → 𝐼, 𝐴𝐸𝐼 → 𝐶 we can know that relation R isn’t in 2NF

Therefore, 1NF is the highest normal form of R.

1. **Find a minimal cover Fm for F. (2 marks)**

Step 1: Reduce right side.

Fm = {AD → B, BD → G, BE → I, AE → D, AE → I, AI → E, AEI → C}

Step 2: Reduce left side.

Fm = {AD → B, BD → G, BE → I, AE → D, AE → I, AI → E, AE → C}

Step 3: Remove redundant FDs.

Fm = {AD → B, BD → G, BE → I, AE → D, AE → I, AI → E, AE → C}

Therefore, the minimal cover Fm = {AD → B, BD → G, BE → I, AE → D, AE → I, AI → E, AE → C}

1. **Regarding F, does the decomposition R1 = {ABCDJ}, R2 = {BDGI}, R3 = {BCEH} of R satisfy the lossless join property? (2 marks)**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | E | G | H | I | J |
|  | a | a | a | a | b | ~~b~~ a | b | b | a |
|  | b | a | b | a | b | a | b | a | b |
|  | b | a | a | b | a | b | a | b | b |

Since there is row that all take the value a, then the decomposition R1, R2, R3 doesn’t satisfy the lossless join property.

1. **Provide a step-by-step lossless decomposition of R into BCNF normal form. (3marks)**

Consider AD → B, in the minimal cover is not a super key. Decompose R into R1 and R2.

R1= {A, D, B} with key: {AD}

R2= {A, D, C, E, G, H, I, J} with AE → DI violates BCNF since AE is not super key.

R11= {A, E, D, I} with key: {AE}

R12= {A, E, C, G, H, J}

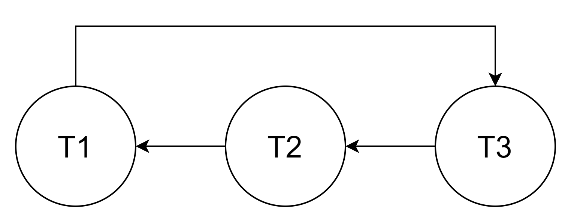
**Question 2 (8 marks)**

Consider the schedule below. Here, R(\*) and W(\*) stand for ‘Read’ and ‘Write’, respectively. T1, T2 and T3 represent 3 transactions, and t1, t2,…, t12 represents different time slots.

Note: Each transaction begins at the time slot of its first operation and commits right after its last operation (same time slot).

**(1)Is the schedule serializable? If it is not serializable, draw a precedence graph; otherwise, provide an equivalent serial schedule. (2 marks)**

It is not serializable.



**(2) Use the Two-Phase Locking protocol to add appropriate locks/unlocks for each of the transactions. (3 marks)**

|  |  |  |
| --- | --- | --- |
| T1 | T2 | T3 |
| read\_lock(x); | read\_lock(y); | read\_lock(z); |
| R(x) | R(y) | R(z) |
| read\_lock(y); | read\_lock(z); | read\_lock(x); |
| R(y) | R(z) | R(x) |
| write\_lock(x); | write\_lock(y); | write\_lock(z); |
| W(x) | W(y) | W(z) |
| write\_lock(y); | write\_lock(z); | write\_lock(x); |
| W(y) | W(z) | W(x) |
| unlock(x); | unlock(y); | unlock(x); |
| unlock(y); | unlock(z); | unlock(z); |

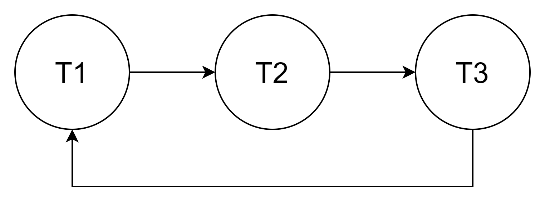
**(3) Based on the locks/unlocks you added in q(2), if the transactions attempt to**

**follow the schedule in the table, what will happen? Please justify your answer**

**using the method learned from the lecture. (3 marks)**

There will be an issue of deadlock.

Create a wait-for graph for currently active transactions:



**Question 3 (6 marks)**

Consider the following page request sequence:

P1, P2, P1, P4, P3, P7, P2, P1, P4, P5, P8, P6, P8, P2, P8.

Assume there are 3 buffers in the buffer pool.

**(1) Sketch the process of how blocks are replaced in the Least Recently Used (LRU)**

**policy, and calculate the cache hit rate. (2 marks)**

|  |  |  |
| --- | --- | --- |
| Q1 | Q2 | Q3 |
| P2 | P1 | P4 |
| P1 | P4 | P3 |
| P4 | P3 | P7 |
| P3 | P7 | P2 |
| P7 | P2 | P1 |
| P2 | P1 | P4 |
| P1 | P4 | P5 |
| P4 | P5 | P8 |
| P5 | P8 | P6 |
| P5 | P6 | P8 |
| P6 | P8 | P2 |

Hit rate= =20%

**(2) Sketch the process of how blocks are replaced in the Most Recently Used (MRU)**

**policy, and calculate the cache hit rate. (2 marks)**

|  |  |  |
| --- | --- | --- |
| Q1 | Q2 | Q3 |
| P1 | P2 | P4 |
| P1 | P2 | P3 |
| P1 | P2 | P7 |
| P1 | P2 | P7 |
| P4 | P2 | P7 |
| P5 | P2 | P7 |
| P8 | P2 | P7 |
| P6 | P2 | P7 |
| P8 | P2 | P7 |
| P8 | P2 | P7 |
| P8 | P2 | P7 |

Hit rate= =33%

**(3) Sketch the process of how blocks are replaced in the First In First Out (FIFO)**

**policy, and calculate the cache hit rate. (2 marks)**

|  |  |  |
| --- | --- | --- |
| Q1 | Q2 | Q3 |
| P1 | P2 | P4 |
| P3 | P2 | P4 |
| P3 | P7 | P4 |
| P3 | P7 | P2 |
| P1 | P7 | P2 |
| P1 | P4 | P2 |
| P1 | P4 | P5 |
| P8 | P4 | P5 |
| P8 | P6 | P5 |
| P8 | P6 | P2 |

Hit rate= =20%