**Database Systems CSE 5331.301, Summer 2020**

**Sample outputs for project 1**

**Rigorous 2 PL concurrency control protocol with wound-wait method for dealing with deadlock**

**Example Input 1:**

b1;

r1 (Y);

r1 (Z);

b2;

r2 (Y);

b3;

r3 (Y);

w1 (Z);

w3 (Y); ​

w2 (Y); ​

r2 (X);

e1;

e3;

w2 (X);

e2;

**Option 1 for displaying your program output:**

**Example Output for Example Input 1:**

**Include Operation in the Beginning and write what happens after every operation. This particular output adds extra explanations – your program output can be more compact:**

b1; Begin T1: Record is added to transaction table with Tid=1 and TS(T1)=1. T1 state=active.

r1 (Y); Y read locked by T1: Lock table record for Y is created with mode R (T1 holds lock).

r1 (Z); Z read locked by T1: Lock table record for Z is created with mode R (T1 holds lock).

b2; Begin T2: Record is added to transaction table with Tid=2 and TS(T2)=2. T2 state=active.

r2 (Y); Y read locked by T2: Lock table record for Y is updated (both T1, T2 hold R lock on Y).

b3; Begin T3: Record is added to transaction table with Tid=3 and TS(T3)=3. T3 state=active.

r3 (Y); Y read locked by T3: Lock table record for Y is updated (T1, T2, T3 hold R lock on Y).

w1 (Z); Read lock upgraded to write lock for item Z by T1, lock table updated to mode W.

w3 (Y);  ​Because Y is read locked by T1, T2 and T3, wound-wait is checked comparing TS(T3) with TS(T1) then TS(T2). Because T3 is younger than T1 and T2, T3 waits on item Y to be released by both T1, T2. Transaction and lock tables updated. List of waiting operations of T3 -> w3 (Y). T3 state=blocked.

w2 (Y);  ​Because Y is read locked by T1, T2 and T3, wound-wait is checked comparing TS(T2) with TS(T1) the TS(T3). T3 is wounded (aborts) as T3 is younger than T2 but T2 is blocked waiting for T1 to release item Y. T3 is aborted and releases R lock from item Y. Transaction and lock tables updated to remove T3 waiting operations. T3 state=aborted, T2 state=blocked. List of waiting operations of T2 -> w2 (Y).

r2 (X); Because T2 is blocked, we add this to list of waiting operation of T2 -> w2 (Y), r2 (X).

e1;  T1 state=committed. Releases all locks held by T1 i.e. Y and Z are unlocked. Checks the wait queues for Y and Z. Resumes T2: T2 state=active, execute list of waiting operations of T2: w2(Y) item Y lock mode is upgraded to W; r2 (X) item X is read locked by T2.

e3;  Already aborted

w2 (X); Item X lock mode is upgraded to W

e2; T2 state=committed. Releases all locks held by T2. i.e. Y and X are unlocked. Checks the wait queues for Y and X: no waiting transactions.

Example Input 2:

b1;

r1(Y);

w1(Y);

r1(Z);

b2;

r2(Y);

b3;

r3(Z);

w1(Z);

e1;

w3(Z);

e3;

e2;

**Option 2 for displaying your program output:**

**Example Output for Example Input 2:**

**This output style shows the Transaction Table and Lock Table after each operation in the input**

Operation = **b1**;

Begin Transaction: T1.

Transaction Table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **T ID** | **TimeStamp** | **State** | **BlockedBY** | **BlockedOperations** |
| T1 | 1 | Active | None | [] |

Lock Table:

|  |  |  |
| --- | --- | --- |
| **DataItem** | **Lock Mode** | **T ID** |
|  |  |  |

Operation **r1 (Y)**;

Item Y is read locked by T1.

Transaction Table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **T ID** | **TimeStamp** | **State** | **BlockedBY** | **BlockedOperations** |
| T1 | 1 | Active | None | [] |

Lock Table:

|  |  |  |
| --- | --- | --- |
| **DataItem** | **Lock Mode** | **T ID List** |
| Y | R | T1 |

Operation **w1(Y)**;

Read lock upgraded to write lock on item Y by T1.

Transaction Table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **T ID** | **TimeStamp** | **State** | **BlockedBY** | **BlockedOperations** |
| T1 | 1 | Active | None | [] |

Lock Table:

|  |  |  |
| --- | --- | --- |
| **DataItem** | **Lock Mode** | **T ID List** |
| Y | W | T1 |

Operation **r1 (Z);**

Data Item Z Read locked by the transaction T1.

Transaction Table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **T ID** | **TimeStamp** | **State** | **BlockedBy** | **BlockedOperations** |
| T1 | 1 | Active | None | [] |

Lock Table:

|  |  |  |
| --- | --- | --- |
| **DataItem** | **Lock Mode** | **T ID List** |
| Y | W | T1 |
| Z | R | T1 |

Operation **b2**;

Begin Transaction: T2

Transaction Table:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T ID** | **TimeStamp** | | **State** | | **BlockedBY** | | **BlockedOperations** | |
| T1 | | 1 | | Active | | None | | [] |
| T2 | | 2 | | Active | | None | | [] |

Lock Table:

|  |  |  |
| --- | --- | --- |
| **DataItem** | **Lock Mode** | **T ID List** |
| Y | W | T1 |
| Z | R | T1 |

Operation **r2 (Y)**;

T2 is blocked by wound-wait as item Y is held by T1 and T2 is younger than T1.

Transaction Table:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T ID** | **TimeStamp** | | **State** | | **BlockedBY** | | **BlockedOperations** | |
| T1 | | 1 | | Active | | None | | [] |
| T2 | | 2 | | Blocked | | T1 | | [r2(Y)] |

Lock Table:

|  |  |  |
| --- | --- | --- |
| **DataItem** | **Lock Mode** | **T ID List** |
| Y | W | T1 |
| Z | R | T1 |

Operation **b3**:

Begin Transaction T3

Transaction Table:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T ID** | **TimeStamp** | | **State** | | **BlockedBY** | | **BlockedOperations** | |
| T1 | | 1 | | Active | | None | | [] |
| T2 | | 2 | | Blocked | | T1 | | [r2(Y)] |
| T3 | | 3 | | Active | | None | | [] |

Lock Table:

|  |  |  |
| --- | --- | --- |
| **DataItem** | **Lock Mode** | **T ID List** |
| Y | W | T1 |
| Z | R | T1 |

Operation **r3 (Z)**;

Item Z is read locked by Txn1 and Txn3.

Transaction Table:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T ID** | **TimeStamp** | | **State** | | **BlockedBY** | | **BlockedOperations** | |
| T1 | | 1 | | Active | | None | | [] |
| T2 | | 2 | | Blocked | | T1 | | [r2(Y)] |
| T3 | | 3 | | Active | | None | | [] |

Lock Table:

|  |  |  |
| --- | --- | --- |
| **DataItem** | **Lock Mode** | **T ID List** |
| Y | W | T1 |
| Z | R | T1, T3 |

Operation **w1 (Z)**:

T3 tries to upgrade R lock on Z to W lock, T3 gets aborted by wound-wait as it is younger than T1. T1 upgrade to W lock on item Z is successful.

Transaction Table:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T ID** | **TimeStamp** | | **State** | | **BlockedBY** | | **BlockedOperations** | |
| T1 | | 1 | | Active | | None | | [] |
| T2 | | 2 | | Blocked | | T1 | | [r2(Y)] |
| T3 | | 3 | | Aborted | | None | | [] |

Lock Table:

|  |  |  |
| --- | --- | --- |
| **DataItem** | **Lock Mode** | **T ID List** |
| Y | W | T1 |
| Z | W | T1 |

Operation **e1:**

Commits T1 and release T1 locks on Y, Z. T2 resumes and executes waiting operations r2(Y), gets R lock on item Y.

Transaction Table:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T ID** | **TimeStamp** | | **State** | | **BlockedBY** | | **BlockedOperations** | |
| T1 | | 1 | | Committed | | None | | [] |
| T2 | | 2 | | Active | | None | | [] |
| T3 | | 3 | | Aborted | | None | | [] |

Lock Table:

|  |  |  |
| --- | --- | --- |
| **DataItem** | **Lock Mode** | **T ID List** |
| Y | R | T2 |

Operation: **w3(Z):**

T3 is already aborted. So, no changes in the tables.

Operation: **e3:**

T3 is already aborted. So no changes in the tables.

Operation **e2:**

Commits T2 and release T2 lock on Y.

Transaction Table:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T ID** | **TimeStamp** | | **State** | | **BlockedBY** | | **BlockedOperations** | |
| T1 | | 1 | | Committed | | None | | [] |
| T2 | | 2 | | Committed | | None | | [] |
| T3 | | 3 | | Aborted | | None | | [] |

Lock Table:

|  |  |  |
| --- | --- | --- |
| **DataItem** | **Lock Mode** | **T ID List** |
|  |  |  |