

CSE 344 Section 7 Worksheet

(a) Consider a concurrency control manager that uses strict two phase locking that schedules three transactions:

- $T1 : R1(A), R1(B), W1(A), W1(B), Co1$
- $T2 : R2(B), W2(B), R2(C), W2(C), Co2$
- $T3 : R3(C), W3(C), R3(A), W3(A), Co3$

Each transaction begins with its first read operation, and commits with the Co statement. Answer the following questions for each of the schedules below:

- Is the schedule conflict-serializable? If yes, indicate a serialization order.
- Is this schedule possible under a strict 2PL protocol?
- If strict 2PL does not allow this schedule because it denies a read or write request, is the system in a deadlock at the time when the request is denied?

i. Schedule 1:

$R2(B), W2(B), R3(C), W3(C), R3(A), W3(A), Co3, R2(C), W2(C), Co2, R1(A), R1(B), W1(A), W1(B), Co1$

- a) Is the schedule conflict-serializable? If yes, indicate a serialization order.
- b) Is it possible under strict 2PL?
- c) Does strict 2PL lead to a deadlock?

ii. Schedule 2:

$R2(B), W2(B), R3(C), W3(C), R1(A), R1(B), W1(A), W1(B), Co1, R2(C), W2(C), Co2, R3(A), W3(A), Co3$

- a) Is the schedule conflict-serializable? If yes, indicate a serialization order.
- b) Is it possible under strict 2PL?
- c) Does strict 2PL lead to a deadlock?

(b) Given the following three transactions:

T1: R(A), W(B), I(D), R(C), Commit

T2: R(B), R(D), W(C), Commit

T3: R(D), R(C), R(D), W(A), Commit

Assume that R(X) reads all tuples in table X, W(X) updates all tuples in X, and I(X) inserts one new tuple in X.

Does there exist a schedule of the above transactions that would result in a deadlock if executed under strict 2PL with **only exclusive table locks**? If so, write such a schedule with lock and unlock operations and indicate why the transactions are deadlocked. Otherwise write "No". Use L1(A) to refer to T1 locking table A, and U1(A) for unlocking.

(c) What indexes could we make on Users?

```
CREATE TABLE Users (  
id INT PRIMARY KEY,  
age INT,  
score INT, ...);
```

Expecting 1000 exec/day

```
SELECT *  
FROM Users, Assets  
WHERE Users.id = Assets.uid
```

Expecting 1000 exec/day

```
SELECT *  
FROM Users  
WHERE Users.score > 95
```

Expecting 10 exec/day

```
SELECT *  
FROM Users  
WHERE Users.age > 21
```