

COMP2411 Fall 2023 Class Exercise 10

Student Name: _____

Student ID: _____

Question 1. Consider the following two transactions:

T1: R(A); R(B); if A=0 then B:=B+2; W(B)
T2: R(B); R(A); if B=0 then A:=A-1; W(A)

Assume the initial values A=B=0, and that we have the consistency requirement: A=0 or B=0.

- A. Show if every serial execution of T1 and T2 maintains consistency.
- B. Provide a non-serializable schedule for T1 and T2.
- C. Given two transactions T1 and T2 consisting of n1 and n2 actions, respectively. How many interleavings (i.e., different schedules with T1 and T2) are possible?

Question 2. Consider a database with three data items X, Y, and Z. Assume there are two transactions T1 and T2 as follows:

T1: R(X); R(Y); W(X); W(Y)
T2: R(Z); R(X); W(X)

Given the following schedule of interleaved operations from T1 and T2:

S: R2(Z); R2(X); R1(X); R1(Y); W1(X); W2(X); C2; W1(Y); C1

“C1” means transaction T1 commits. Answer each of the following questions:

- A. What type of anomaly occurs in this schedule S? Explain your answer.
- B. For each of T1 and T2, insert all the lock and unlock operations to make the transaction satisfy the strict 2PL protocol.
- C. Show that modifying the schedule according to strict 2PL can prevent the anomaly. Justify your answer.

Answers for Question 1:

A. Execute T1 then T2:

Initially: A=B=0

After T1: A=0; B=B+2=2

After T2: A=A=0; B=2

Execute T2 then T1:

Initially: A=B=0

After T2: A=A-1=-1; B=0

After T1: A=-1; B=B=0

Therefore, every serial execution of transactions T1 and T2 maintains consistency.

B. Below is an example of the non-serializable schedule for T1 and T2:

Time	T1	T2
1	R(A)	
2	R(B)	
3		R(B)
4		R(A)
5	if A=0 then B = B+2	
6		if B=0 then A = A-1
7	W(B)	
8		W(A)

Initially: A=B=0

After executing the above schedule: A=A-1=-1; B=B+2=2

Therefore, the schedule is non-serializable.

C. The number of possible schedules = $C_{n_1+n_2}^{n_1}$ or $C_{n_1+n_2}^{n_2}$

Answers for Question 2:

A. Lost update anomaly. T2 has overwritten the uncommitted update of X by T1.

B. T1: WL(X); R(X); WL(Y); R(Y); W(X); W(Y); UL(X); UL(Y)
T2: RL(Z); R(Z); WL(X); UL(Z); R(X); W(X); UL(X)

C. In strict 2PL, T1 cannot get an exclusive lock on X because T2 is already holding an exclusive lock on X which would not be released until T2 commits. The modified schedule would be S': R2(Z); R2(X); W2(X); C2; R1(X); R1(Y); W1(X); W1(Y); C1