

Solusi Latihan PPT Concurrency Control

9.1. Two Phase Locking & Deadlock.....	2
a. Two Phase Locking.....	2
b. Deadlock.....	3
9.2. Legal Two-phase Locking.....	4
Problem.....	4
Solution.....	4
9.3. Automatic Acquisition of Lock.....	5
Problem.....	5
Solution.....	5
9.4. Deadlock Handling.....	6
Problem.....	6
Solution.....	7
a. Wait-die.....	7
b. Wound-wait.....	8

9.1. Two Phase Locking & Deadlock

Exercise 9.1

Consider the following two transactions:

T1: **read (A)**
 read (B)
 B := B + 0.1*A
 write (B)

T2: **read (B)**
 read (A)
 A := A - 0.05*B
 write (A)

- a) Add lock and unlock instructions to both transactions so that they follow the two-phase locking protocol.
- b) Can the execution of these two transactions result in a deadlock?





Solution

a. Two Phase Locking

Exercise 9.1 – Solution (a)

Add lock and unlock instructions to both transactions so that they follow the two-phase locking protocol.

T1: lock-S (A)	T2: lock-S (B)
read (A)	read (B)
lock-X (B)	lock-X (A)
read (B)	read (A)
B := B + 0.1*A	A := A - 0.05*B
write (B)	write (A)
unlock (A)	unlock (B)
unlock (B)	unlock (A)

©Silberschatz et.al. (2019)
[modified by Tim Pengajar IF3140 Semester 1 2024/2025]

b. Deadlock

22

Exercise 9.1 – Solution (b)



Can the execution of these two transactions result in a deadlock?

T1:	T2:	CC Manager
lock-S (A)		grant-S (A, T1)
read (A)		
	lock-S (B)	grant-S (B, T2)
	read (B)	
lock-X (B)		queue-X (B, T1) – T1 wait for T2
	lock-X (A)	queue-X (A, T2) – T2 wait for T1

deadlock

[Back](#)

©Silberschatz et al. (2019)
[modified by Tim Pengajar IF3140 Semester 1 2024/2025]



9.2. Legal Two-phase Locking



Problem

15

Exercise 9.2

Is the following schedule a two-phase locking (2PL) schedule (legal under 2PL protocol)?

R1(A); R2(A); R3(B); W1(A); R2(C); R2(B); W2(B); W1(C);

Solution
©Silberschatz et.al. (2019)
[modified by Tim Pengajar IF3140 Semester 1 2024/2025]

Solution

23



Exercise 9.2 – Solution

R1(A); R2(A); R3(B); W1(A); R2(C); R2(B); W2(B); W1(C);

Add lock and unlock instructions to the schedule:

- T1 reads and then writes A → needs exclusive lock on A before R1(A)
LX1(A); R1(A); R2(A); R3(B); W1(A); R2(C); R2(B); W2(B); W1(C);
- T2 reads (only) A → needs shared lock on A before R2(A). Since A is locked exclusively by T1, shared lock on A by T2 can not be granted.

Therefore, the schedule is not a 2PL schedule.

  [Back](#)

9.3. Automatic Acquisition of Lock

Problem

Exercise 9.3

Instructions from T1, T2, and T3 arrive in the following order.

R1(A); R2(A); R3(B); W1(A); R2(C); R2(B); C3; W2(B); C2; W1(C); C1;

What is the final schedule if the 2-phase locking with automatic acquisition of locks is implemented by CC Manager?

Solution
©Silberschatz et al. (2019)
[modified by Tim Pengajar IF3140 Semester 1 2024/2025]

Solution

*disclaimer on

⇒ **LS1(A); R1(A); LS2(A); LS3(B); LS2(C); LS2(B); C3; UL(B); LX2(B); C2; UL(B); UL(C); UL(A); LX1(A); LX1(C); C1; UL(X); UL(A)**

⇒ Sebenarnya sama aja kayak yang 9.4. wait-die

9.4. Deadlock Handling

Problem

Exercise 9.4 – Deadlock Prevention

Instructions from T1, T2, and T3 arrive in the following order (the same as Exercise 9.3).

R1(A); R2(A); R3(B); W1(A); R2(C); R2(B); C3; W2(B); C2; W1(C); C1;

What is the final schedule if the 2-phase locking with automatic acquisition of locks with

- a. wait-die deadlock prevention scheme
- b. wound-wait deadlock prevention scheme

is implemented by CC Manager?

Assume that Timestamp (T1,T2,T3)=(1,2,3)

Solution

a. Wait-die

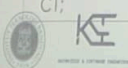
3

Exercise 9.4 – Solution (a. wait-die)

T1	T2	T3	CC Manager / Remark
R1(A);			grant-S (A,T1)
	R2(A);		grant-S (A,T2)
		R3(B);	grant-S (B,T3)
W1(A);			queue-X (A,T1)
	R2(C);		grant-S (C,T2)
	R2(B);		grant-S (B,T2)
		C3;	unlock (B,T3)
	W2(B);		grant-X (B,T2)
	C2;		unlock (A,T2); unlock (B,T2); unlock (C,T2)
W1(A);			grant-X (A,T1)
W1(C);			grant-X (C,T1)
C1;			

Older Transaction will wait for locks held by new transactions

Final Schedule (without locks):
R1(A); R2(A); R3(B); R2(C); R2(B); C3; W2(B); C2; W1(A); W1(C); C1;



©Silberschatz et.al. (2019)
(modified by Tim Pengajar IF3140 Semester 1 2024/2025)

Credit to Evelyn Yosiana

b. Wound-wait

32

Exercise 9.4 – Solution (b. wound-wait)

T1	T2	T3	CC Manager / Remark
R1(A);		grant-S (A,T1)	
	R2(A);	grant-S (A,T2)	Older Transaction will wound new transactions; T2 was rolled back
		grant-S (B,T3)	
	A2;	unlock(A,T2)	Before it is granted, all following T2 instructions will be queued.
W1(A);	R2(A);	grant-X (A,T1)	
		queue-S(A,T2)	
		unlock (B,T3)	
	C3;	grant-X (C,T1)	
W1(C);		unlock (A,T1); unlock (C,T1)	
C1;		grant-S (A,T2)	
	R2(A);	grant-S (C,T2)	
	R2(C);		
	R2(B);		
	W2(B);		
	C2;		

Final Schedule (without locks):
R1(A); R2(A); R3(B); A2; W1(A); C3; W1(C); C1; R2(A); R2(C); R2(B); W2(B); C2;

©Silberschatz et al. (2011)
 (modified by Tim Pengajar IF3140 Semester 1 2024) [Back](#)

Credit to Evelyn Yosiana