SOLUTIONS

Question 3 revised 10:15 pm on April 4

Page 3 revised 3:30 pm on March 30

CS 425/ECE 428 Distributed Systems
Spring 2018

Homework 7

Total points: 30

Due: by 7 p.m. April 5, 2018 (Thursday)

2 submission attempts permitted for this homework.

1. (10 points) Transactions T and U execute on a single server. In the table below, time proceeds from top to bottom, and the relative position of the operations indicate their relative order in time. Assume that a different lock is used for each variable w, x, y and z.

Assume that two-phase locking is used, and that each transaction acquires locks only when needed. For instance, in part (i) below, transaction T acquires lock for variable z right before it performs read(z). All locks held by a transaction are released when it performs closeTrasaction.

Determine whether the interleaving of operation as shown in parts (i) and (ii) below can occur under the assumption stated in each part (state YES or NO).

Notation: read (x) returns the value of variable x, and write(x,1) writes 1 to variable x.

(i) ***Assume that Exclusive locks are used with two phase locking***: ___NO_____

2 points for NO

If you answer NO, **delete** a minimum number of operations from the transactions so that the modified execution becomes possible while using exclusive locks. Assume that the relative order of remaining operations stays unchanged, even if some of the operations are deleted.

3 points for deleting a correct operation

Transaction T	Transaction U
openTransaction	
	openTransaction
	b = read (y)
a = read (x)	
write (w,2)	
	c = read (x)
	write (z,1)
	write (y,6)
	closeTransaction
d = read (y)	
closeTransaction	

Deleting any one of the highlighted operations suffices

(ii) ***Assume that Read-Write locks are used with two phase locking***: ___YES_____

(With read-write locking, a transaction acquires *read lock* when that is adequate. The transaction may promote a *read lock* to a *write lock* later if needed.)

5 points for YES

0 points for NO

If you answer NO, **change** a minimum number of **write** operations to **read** operations so that the modified execution becomes possible while using read-write locks.

See the execution on the next page.

Transaction T	Transaction U
openTransaction a = read (x) e = read (y) write(x,5) write (y,2) closeTransaction	 openTransaction c = read (z) write (z,1) f = read(y) closeTransaction

Question 2 is on the next page.

2. (10 points)

(a) The table below shows the interleaving of the operations performed by transactions T, U and V. Suppose that optimistic concurrency control with *forward* validation is used. Validation is carried out when a transaction performs closeTransaction operation. When forward validation fails, **assume that the transaction performing the validation is immediately aborted**. Which of the transactions (if any) must be aborted?

Transaction T	Transaction U	Transaction V
openTransaction		
	openTransaction	
		openTransaction
a = read (w)		
write (y,1)		
	e = read(y)	
		c = read (v)
		write (w,5)
		closeTransaction
write (z,5)		
	d = read (w)	
	write (w,5)	
	closeTransaction	
b = read (w)		
write (v,5)		
closeTransaction		

-3 for each incorrect transaction abort

U and V are aborted.
T is not aborted.

Question 2, part (b) and Question 3 are on next page.

(b) The table below shows the interleaving of the operations performed by transactions T1, T2 and T3.

Suppose that **backward** validation is used. Which of the transactions (if any) must be aborted.

T1	**T2**	**T3**
	openTransaction	
openTransaction	write(a)	
read(b)		
write(c)		
closeTransaction	read(b)	
	read(a)	
		openTransaction
		write(a)
	write(c)	read(c)
	closeTrasaction	
		closeTransaction

-3 points for each incorrect abort.

T3 is aborted.
T1 and T2 are not aborted.

3. (10 points)

part a - 3 points
part b - 3 points
part c - 4 points

Consider a network that consists of 5 nodes A, B, C, D and E. Nodes A and E are connected to B, C and D with bidirectional links. There are no other links in the network. Suppose that node A is able to sign its messages (using its ~~public~~ private key). Assume that a signed message cannot be tampered (i.e., modified) by any node that does not know the signers private key without the tampering being detected. Assume that nodes B, C, D, E know A's public key, but not its private key.

It is assumed that tampering of signed message can be detected by the recipient. Assume that A sends copies of each message to E via A, B, C,

(a) Suppose that at most two of the nodes B, C and D suffer Byzantine failure. Is it possible for node A to send a message to node E reliably? State YES or NO, and explain briefly.

YES. At least one of the copy will be received without tampering.

(b) Suppose that all three nodes, B, C and D suffer Byzantine failure? Is it possible for node A to send a message to node E reliably? State YES or NO.

NO. Since all copies will be tampered.

(c) Answer part (b) again, but this time assuming that nodes B, C and D also know A's private key.

NO.