

Problem #1 (10 points)

Shown below is a concurrent schedule S of four transactions operating under a locking version of concurrency control, determine if S is serializable. If S is serializable produce a serial schedule which is equivalent to S. If a cycle exists, list the nodes in the cycle.

S = [(T2: Slock A) (T3: Slock A) (T2: Xlock B) (T2: Unlock A) (T3: Xlock A)
 (T2: Unlock B) (T1: Slock B) (T3: Unlock A) (T4: Slock B) (T1: Slock A)
 (T4: Unlock B) (T1: Xlock C) (T1: Unlock A) (T4: Xlock A) (T4: Unlock A)
 (T1: Unlock B) (T1: Unlock C)]

Problem #2 (10 points)

Shown below is a concurrent schedule S of five transactions operating under a locking version of concurrency control, determine if S is serializable. If S is serializable produce a serial schedule which is equivalent to S. If a cycle exists, list the nodes in the cycle.

S = [(T1: Xlock A) (T2: Xlock B) (T1: Unlock A) (T3: Xlock A) (T4: Xlock C)
 (T4: Xlock D) (T2: Unlock B) (T3: Unlock A) (T3: Xlock B) (T4: Xlock E)
 (T5: Xlock A) (T3: Unlock B) (T2: Xlock B) (T2: Unlock B) (T4: Unlock C)
 (T4: Unlock D) (T4: Unlock E) (T5: Unlock A) (T4: Xlock A) (T4: Unlock A)
 (T4: Xlock B) (T4: Unlock B)]

Problem #3 (10 points)

Shown below is a concurrent schedule S of three transactions operating under a timestamp version of concurrency control, the timestamp of each transaction, and the initial read and write timestamp values for all of the objects referenced in S. Using the timestamping protocol, determine the action at each time instance of S that the protocol will take with respect to the transaction attempting the operation.

ts(T1) = 20	ts(T2) = 30	ts(T3) = 10
rts(A) = 10	rts(B) = 15	rts(C) = 0
wts(A) = 5	wts(B) = 0	wts(C) = 0

S = [(T1: read A) (T3: read C) (T1: write B) (T2: write A) (T2: write C) (T3: write C)
 (T3: write A) (T1: write B)]

Problem #4 (10 points)

Using a timestamping technique to prevent deadlock from occurring we presented two different protocols: "wait or die" and "wound or wait". Given the transaction timestamps: $ts(T1) = 10$, $ts(T2) = 5$, $ts(T3) = 12$, and $ts(T4) = 6$, determine the action of both protocols for each of the following scenarios.

- (a) T1 requests an object whose lock is held by T3
- (b) T2 requests an object whose lock is held by T1
- (c) T4 requests an object whose lock is held by T2
- (d) T4 requests an object whose lock is held by T3