EECS 495
Introduction to Database Systems
Fall 2018
Instructor: Mas-ud Hussain
Homework Assignment No. 3
Due: Sunday, November 18, 2018

Problem No. 1:
You are given three transaction schedules below showing the concurrent execution two or three transactions (T1, T2, and T3). These schedules show only the reads and writes of four database records (A, B, C, and D); assume that there is arbitrary computation between these accesses. The system acquires necessary locks on records before accessing them, and releases them according to the rules of two-phase locking.

For each of the following transaction schedules (S1, S2, S3), answer the three questions below indicating whether the schedule is serializable and whether each could have been generated by a database system running two-phase locking as described above (also please briefly explain your reasoning).

(a) Is this schedule permissible by two-phase locking?
(b) Is this schedule view serializable?
(c) If so, what is a serial equivalent ordering of transactions?

T1	T2
Read(A)	
Write(A)	
	Read(B)
	Write(B)
Commit	
	Read(A)
	Commit

Schedule: S1

T1	T2
Read(A)	
Read(B)	
	Read(A)
	Write(A)
Read(C)	
Commit	
	Commit

Schedule: S2

T1	T2	T3
Write(A)		
		Write(A)
	Write (A)	
	Read(B)	
Read(C)		
		Read(D)
Commit		
	Commit	
		Commit

Schedule: S3

Problem No. 2:
Consider the following sequences of actions, listed in the order they are submitted to the DBMS:

- **Sequence S1:** T1:W(X), T2:R(Y), T2:W(X), T1:W(Y), T3:W(Y), T2:R(Y), T1:Commit, T2:Commit, T3:Commit
- **Sequence S2:** T1:W(X), T2:W(X), T2:R(X), T1:W(X), T3:W(Y), T1:R(X), T1:Commit, T2:Commit, T3:Commit

For each sequence above, describe how each of the concurrency control mechanisms below handles the sequence. Assume that the timestamp of transaction Ti is i. For lock-based concurrency control mechanisms, add lock and unlock requests to the previous sequence of actions as per the locking protocol. The DBMS processes actions in the order shown. If a transaction is blocked, assume that all its actions are queued until it is resumed; the DBMS continues with the next action (according to the listed sequence) of an unblocked transaction. When multiple transactions are aborted, we assume they are resubmitted when the executing transactions are all completed. The relative order of their actions follow that of the original sequence.

(a) Strict 2PL with deadlock detection. (draw the waits-for graph at the point of deadlock, if any)
(b) Strict 2PL with timestamps used for deadlock prevention. (Show for wait-die only)
(c) Rigorous (i.e., with locks held until end-of-transaction) 2PL.

Problem No. 3:

Consider a buffer pool with 3 pages that receives requests for the following page numbers:

1, 2, 3, 4, 2, 1, 2, 5, 3, 1

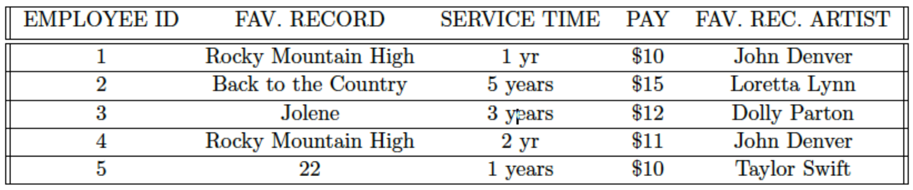
Calculate the hit rate (# of hits / all page requests) for the following replacement policies:

(a) MRU (Most Recently Used)
(b) Clock Replacement

Which strategy do you recommend for this workload?

Problem No. 4:

In order to encourage more employees to participate in the mural-making, the store manager starts a database schema competition, with the winner receiving a commemorative fish statue known as the "*Data Bass*". Since your E-R diagram was so successful, the manager asks you to review the first submission, which collected data from all store employees. For the given table, answer the questions below.



EMPLOYEE ID	FAV. RECORD	SERVICE TIME	PAY	FAV. REC. ARTIST
1	Rocky Mountain High	1 yr	$10	John Denver
2	Back to the Country	5 years	$15	Loretta Lynn
3	Jolene	3 years	$12	Dolly Parton
4	Rocky Mountain High	2 yr	$11	John Denver
5	22	1 years	$10	Taylor Swift

(a) Is the schema for this table in Boyce-Codd Normal Form? Explain why or why not.
(b) Decompose the schema into BCNF, if necessary. Show your work, including the functional dependency you use at each step of the decomposition.