1   Undo Logging Q1

The following table gives the operations for Transaction T in the first column. Disk A = 7 and Disk B = 3 at the beginning.

Operation	Mem A	Mem B	Disk A	Disk B	UNDO Log
READ(A)					
READ(B)					
WRITE(A, A+B)					
WRITE(B, A-B)					
FLUSH(A)					
FLUSH(B)					
COMMIT					

(a) Fill in the columns Mem A, Mem B, Disk A, Disk B.

(b) If the system crashes right before COMMIT, how will we recover? Fill in the UNDO column.

(c) What happens if the system crashes again while we're undoing? How do we recover?

2 Redo Logging Q1

The following table gives the operations for Transaction T in the first column. Disk A = 5 and Disk B = 4 at the beginning.

Operation	Mem A	Mem B	Disk A	Disk B	REDO Log
READ(A)					
READ(B)					
WRITE(A, A+B)					
WRITE(B, A-B)					
COMMIT					
FLUSH(A)					
FLUSH(B)					

(a) Fill in the columns Mem A, Mem B, Disk A, Disk B.

(b) If the system crashes right after COMMIT, how will we recover? Fill in the REDO column.

3 Recovery Q1

Consider the execution of the ARIES recovery algorithm given the following log (assume a checkpoint is completed before LSN 0 and the Dirty Page Table and Transaction Table for that checkpoint are empty):

LSN	Log Record
10	update: T1 writes P1
20	update: T2 writes P3
30	T1 commit
40	update: T3 writes P4
50	update: T2 writes P1
60	T1 end
70	update: T3 writes P2
80	T2 abort

(a) During Analysis, what log records are read? What are the contents of the transaction table and the dirty page table at the end of the analysis stage?

(b) During Redo, what log records are read? What data pages are read? What operations are redone (assuming no updates made it out to disk before the crash)?

(c) During Undo, what log records are read? What operations are undone? Show any new log records that are written for CLR's. Start at LSN 100. Be sure to show the undoNextLSN.

4 Recovery Q2

Your database server has just crashed due to a power outage. You boot it up, find the following log and checkpoint information on disk, and begin the recovery process. Assume we use a STEAL/NO FORCE recovery policy.

LSN	Record	prevLSN
30	update: T3 writes P5	null
40	update: T4 writes P1	null
50	update: T4 writes P5	40
60	update: T2 writes P5	null
70	update: T1 writes P2	null
80	begin_checkpoint	-
90	update: T1 writes P3	70
100	end_checkpoint	-
110	update: T2 writes P3	60
120	T2 commit	110
130	update: T4 writes P1	50
140	T2 end	120
150	T4 abort	130
160	update: T5 writes P2	null
180	CLR: undo T4 LSN 130	150

Transaction Table			Dirty Page Table	
Transaction	lastLSN	Status	PageID	recLSN
T1	70	Running	P5	50
T2	60	Running	P1	40
T3	30	Running		
T4	50	Running		

(a) The log record at LSN 60 says that transaction 2 updated page 5. Was this update to page 5 successfully written to disk? The log record at LSN 70 says that transaction 1 updated page 2. Was this update to page 2 successfully written to disk?

(b) At the end of the analysis phase, what transactions will be in the transaction table, and what pages will be in the dirty page table?

(c) At which LSN in the log should redo begin? Which log records will be redone (list their LSNs)? All other log records will be skipped.