

Problem Set: Recovery from Failures

DBMS Course

Problem 1. Suppose that the consistency constraint on the database is $0 \leq A \leq B$. State whether each of the following transactions preserves consistency:

1. $A := A + B; B := A + B;$
2. $B := A + B; A := A + B;$
3. $A := B + 1; B := A + 1;$

For each of the transactions above, add the read and write-actions to the computation and show the effect of the steps on main memory and disk. Assume initially $A = 5$ and $B = 10$. Also, state whether it is possible, with the appropriate order of **OUTPUT** actions, to assure that consistency is preserved even if there is a crash while the transaction is executing.

Undo Logging

Problem 2. Show the undo log records for each of the transactions in the previous problem (call each transaction T), assuming initially $A = 5$ and $B = 10$.

Problem 3. For each of the sequences of log records representing the actions of one transaction T , state all the sequences of events that are legal according to the rules of undo logging, where the events of interest are the writing to the disk of the blocks containing datanase elements, and the blocks of the log containing the update and commit records. You may assume that logi records are written to the disk in the order shown; that is, it is not possible to write one log record to disk while a previous record is not written to disk.

1. $\langle \text{START } T \rangle; \langle T, A, 10 \rangle; \langle T, B, 20 \rangle; \langle \text{COMMIT } T \rangle;$
2. $\langle \text{START } T \rangle; \langle T, A, 10 \rangle; \langle T, B, 20 \rangle; \langle T, C, 30 \rangle; \langle \text{COMMIT } T \rangle;$

Problem 4. The following is a sequence of undo-log records written by two transactions T and U :
 $\langle \text{START } T \rangle; \langle T, A, 10 \rangle; \langle \text{START } U \rangle; \langle U, B, 20 \rangle; \langle T, C, 30 \rangle; \langle U, D, 40 \rangle; \langle \text{COMMIT } U \rangle; \langle T, E, 50 \rangle; \langle \text{COMMIT } T \rangle;$

Describe the action of the recovery manager, including changes to both disk and the log, if there is a crash and the last log record to appear on disk is:

- (a) $\langle \text{START } U \rangle$ (b) $\langle \text{COMMIT } U \rangle$ (c) $\langle T, E, 50 \rangle$ (d) $\langle \text{COMMIT } T \rangle$

Problem 5.

For each of the situations described in the previous exercise, what values written by T and U *must* appear on disk? Which values *might* appear on disk?

Problem 6. Suppose that the transaction U in the above problem is changed so that the record $\langle U, D, 40 \rangle$ becomes $\langle U, A, 40 \rangle$. What is the effect on the disk value of A if there is a crash at some point during the sequence of events? What does this example say about the ability of logging by itself to preserve atomicity of transactions?