SAMPLE Final Examination

Time: 50 minutes
Maximum Points: 100

STUDENT NAME: _____

General Instructions:

- This test is a *closed book exam.* You can make use of a one-page cheat sheet.
- *Make sure to show all your work and to state any assumptions you make explicitly!*

PROBLEMS:	MAXIMUM SCORE	YOUR SCORE
problem 1:	10	
problem 2:	30	
problem 3:	15	
problem 4:	30	
problem 5:	15	
total:	100	

Problem 1: Concepts and Their Definitions [10pts] Select the most appropriate definition number for each of the concepts below and mark the number next to the concept.

1. A mechanism to place markers into the log so that transactions can be undone savely.
2. A mechanism to deal with system failure.
3. A mechanism to assure that transactions execute maximally in parallel.
4. Whenever transactions have to wait for a resource, such as a lock, held by another transaction.
5. A mechanism to restore the database to a consistent state.
6. A cycle of waits for resources allowing none of the transactions involved in this cycle to make any more progress.
7. A sequence of actions from one or more transactions
8. A mechanism used when resolving a deadlock problem caused by two transactions needing to access each others resources.
9. A mechanism to assure that no records prior to a certain time will be needed during recovery.
10. A mechanism to assure that transactions execute atomically or not at all.
11. A sequence of read and write actions from one or more transactions.
12. A mechanism to assure that concurrently operating transactions preserve consistency of the database.
13. A mechanism of storing all important actions of a transaction onto a separate location, that then is backed up to disk.

For each concept below, mark the number of the most appropriate definition from above that best describes it.

- Logging.
- Abort.
- Checkpointing.
- Deadlock.
- Concurrency Control.

Problem 2: SHORT-ANSWER Questions on Transactions Management [30pts]

1. Properly implemented transactions are commonly said to meet the ACID test, where ACID stands for the terms A = atomicity, C = consistency, I = isolation, and D = durability. Explain the meaning of the four terms, and how, if at all, each of them is assured?

2. Pick one concurrency control strategy (scheduler), and explain briefly how it works. Indicate if it is a pessimistic or an optimistic strategy.

3. Explain the difference between the three types of schedules: serial, serializable and conflict-serializable. Give a simple example for each.

4. Draw the precedence graph for the schedule below. Is the schedule below conflict-serializable? If yes, list a serial schedule. If no, explain why not.
   $R_1(A)$ $R_1(B)$ $R_2(A)$ $W_3(C)$ $W_2(B)$ $W_2(C)$ $R_4(C)$ $W_1(C)$

5. For two given schedules with identical precedence graphs, can we infer that they are conflict-serializable? Justify your answer.

Problem 3: Relational Algebra Expressions [15pts]

Let R and S be two relations. Cond1 and cond2 could be any valid condition on their relational input argument. are the following relational algebra expressions are correct or not?

1. Is SELECT_[cond1 and cond2] (R) =
   SELECT_[cond1] (SELECT_[cond2] (R)).

2. Is SELECT_[cond1 and cond2] (R JOIN S) =
   SELECT_[cond1] (SELECT_[cond2] (S) JOIN SELECT_[cond1 and cond2] (R)).

3. Is SELECT_[cond1 or cond2] (R JOIN S) =
   (SELECT_[cond1 or cond2] (R) JOIN SELECT_[cond1 or cond2] (S)).

Problem 4: Join Processing [30pts]

Given two relations R(a,b) and S(b,c), compute the IO cost of the query (R NATURAL-JOIN S). **You need NOT compute the cost of (S NATURAL-JOIN R)!**.
Assume the following data:
T(R) = 10,000 = # of tuples of R tuples
block(R) = at most 10 tuples fit onto one block
T(S) = 5,000 = # of tuples of R tuples
block(S) = at most 5 tuples fit onto one block
M = # of blocks in main memory
HT(i) = 3 # of levels in index i
LB(i) = # of leaf blocks in index i
V(R, a) = 40 and V(R, b) = 50 and V(S, b) = 10 and V(S, c) = 30.

1. What are the IO costs of an **index-based join**, when we have B+ indexes R.b and S.b. The indices R.B and S.b are both non-clustered indices and don't fit into main memory? R and S are both not contiguously stored.

2. What are the IO costs of an **index-based join**, when we have B+ indexes R.b and S.b. Only one of the two indices could fit into main memory at one time. The index R.B is a clustered and the index S.b is a non-clustered index. S is not contiguous.

3. What are the IO costs of an **index-based join**, when we have B+ indexes R.b and S.b. Both indices fit in main memory at the same time and both are clustered.

Problem 5: Query Tree Evaluation [15pts]

Let us assume all facts and costs as given in Problem 4.1. In addition, let us assume we have a third relation T(c,d) with:
T(T) = 100,000 = # of tuples of T tuples
block(T) = 100 tuples fit onto one block
V(T,c) = 500 and V(T,d) = 100.
Buffer size M = 100.

Calculate the total estimated IO cost of executing this *one* query plan:
((R NATURAL-JOIN S) NATURAL-JOIN T)