CS 631: ITDBMS: End Semester Exam, 26 April 2012

Time: 2.00 PM – 5.00 PM Marks: 100

Instructions: Answer all parts of a question together.

1. Short answers

   (a) In RAID 1 organization, how would you detect if only one copy of a block had been written when the system failed in the case of (a) hardware RAID and (b) software RAID? ...5

   (b) Explain why an (in-memory) translation table is used for flash storage access (as part of the flash translation layer), whereas it is not needed for magnetic disk access. ...5

   (c) Suppose you have an SQL query of the form
   `SELECT r.A, count(r.B) FROM r WHERE r.C < 10 GROUP BY r.A HAVING sum(r.B) > 100`
   Explain how you could evaluate the above query using map-reduce operations. You do not need to give pseudocode, just give the intuition. ...5

   (d) Very briefly explain why recovery with physiological logging requires atomic page writes, whereas physical logging does not. ...5

   (e) In ARIES, every redo-only compensation log record has an UndoNextLSN field, which allows undo to skip over physical log records that have already been undone. Explain why without this feature, the log may potentially grow very big in case of repeated failures during recovery, and why with this feature, the growth is bounded. (Without this feature, IBM actually encountered a situation where the log disk became full during recovery with repeated crashes during recovery, preventing the system from completing recovery.) ...5

   (f) In a highly distributed data storage system for an OLTP workload, it is impractical to perform distributed joins at run time, whereas it is done routinely for analytical (decision support) workloads. Briefly explain why. ...5

2. Query processing and optimization

   (a) Suppose I have a query
   `SELECT * FROM r, s WHERE r.B=s.B ORDER BY r.A LIMIT 10`
   where the limit clause specifies the number of rows to be retrieved.

      i. If r had an index on r.A and s had an index on s.B, and most r.B values matched s.B values, what would most likely be a good plan for this query? ...3

      ii. If there are no indices, an option is to estimate a value v for r.A such that it is likely that r.A values below v are sufficient to generate 10 results, and run the query after adding a selection $r.A <= v$. (If there are fewer than 10 results, we revise the estimate for v and try again, till we get 10 results.
      Explain how to use standard statistics information (number of distinct values, histograms) to estimate the value v. Also suggest at least one reasonable way to choose a revised value for v, in case we get less than 10 results. ...5+2

   (b) Give a transformation rule, with associated conditions, which allows a join to be eliminated from a query. ...5

   (c) Suppose I wish to compute the natural join of $r(A, B), s(A, C)$ and $t(A, D)$. Give an example to show how a multiway merge-join may perform better in some cases than any fixed order of two-way joins. ...5

3. Benchmarking: Suppose a system runs transaction type A at 50 transactions per second (tps), and type B at 10 tps.

   (a) Given an equal mix of transactions of each type, what is the effective throughput of the system (in tps), assuming no contention? ...2

(b) Now if there are 2 transactions of type A for each transaction of type B, what is the effective throughput of the system? ...2

(c) What will be the effective throughput in case (a) above, if both types of transactions get an exclusive lock on a particular tuple T1 at the beginning of the transaction, and release it at the very end? ...2

4. R-trees

(a) Using the big O notation, give the complexity of the following operations on an R-tree, with a one-line explanation for each, assuming objects are rectangles, and there are a total of n objects in the tree: (i) time for search for objects overlapping a given point, (ii) time for insertion of an object (iii) space used by the tree. ...5

(b) Assuming we have functions inside(x, y, boundingbox), and each R-tree node has fields node.isleaf, node.numchildren, node.child[MaxChildren] and node.childBB[MaxChildren], give pseudocode for a recursive procedure to print all objects that overlap a given point (x,y), assuming objects are rectangles. For leaf nodes, the child pointers point to objects, and you can call object.print() to print out the object. ...5

(c) Assume we have functions dist(x, y, object), distlb(x, y, boundingbox), distub(x, y, boundingbox). Give pseudocode for a procedure findNearest(treenode, x, y), using global variables `nearestDist` and `nearestObjectSet`, to find nearest neighbours (there may be more than one if they are equidistant) of the given point (x,y). ...7

5. Serializability

(a) Give an example schedule with inserts and reads exhibiting non-serializability due to the phantom phenomenon. ...3

(b) Modify the above example to perform an update instead of an insert, while still allowing the phantom phenomenon to occur. ...3

(c) Show what locks would be obtained by next-key index locking in each of the above cases, and explain how it would prevent the above non-serializable schedules from occurring. ...6

6. Snapshot isolation

(a) Give an example of a non-serializable schedule with snapshot isolation due to write-skew. ...3

(b) Give an example of insertion skew with snapshot isolation, which would be prevented by a primary key constraint. ...3

(c) What would happen in the above example, if the code to check the integrity constraints also ran on a snapshot view? ...2

(d) Suppose every item which is read is also updated by each of a given set of transactions. Is write skew possible if this set of transactions is run using snapshot isolation? Explain your answer. ...3

(e) Explain how version numbering can be used to check if a concurrent transaction has updated a given data item (this test is done during commit processing for snapshot isolation). ...4

Total Marks = 100