CS632 Endsem Exam, Spring 2009, Apr 19, 2009, 9.30 AM -12.30 PM

Question paper pages: 3 Total Marks: 100

NOTE: Answer all subparts of a question together, do **not** split them up.

1. Short answers

   (a) Consider the XML structural join paper. Suppose I need to compute a path expression $a//b/c$, instead of just the two-level path expressions discussed in the paper. Suggest two different approaches to computing such an expression. You don't have to give a detailed algorithm, just a clear explanation of how to do it. ...5

   (b) Consider the GPU computation paper. What is the complexity (in big O notation) of their technique for computing Sum? Compare it with the obvious linear algorithm, and explain why their technique may still make sense. ...3+2

   (c) In the "Query Processing for Updates" paper, tuples in the delta relation for updates are split into two, one for delete and one for insert. This split delta operation on an index can be sorted on (locator, action) or (action, locator), where action is delete or insert, and locator is the index key, and then applied on the index. (a) what should the sort order be for the action, i.e. insert first or delete first, and why, and (b) which of the two is likely to be more efficient? ...2+3

   (d) The Pig Latin paper introduces the idea of a co-group, which can group more than one input together. Suppose you extend SQL minimally by adding an aggregate operator called multiset, which simply create a multiset of values in the group. Show how to express "COGROUP results BY queryString, revenue BY queryString", using SQL, where the schemas are results(queryString, url, position), and revenue(queryString, adSlot, amount). ...5

   (e) Inferring authorization is critical for the semantics of the non-Truman model, whereas it is merely an aid to optimization when finding safe plans (under the Truman model, on which the Kabra et al paper is based). Explain. ...5

   (f) Stochastic testing: Suppose that you wish to test not only queries, but also SQL updates. In this case, you have to ensure that the updated relations are the same on two or more databases. Suggest how to do so efficiently, by using a materialized view along with tuple checksums. ...5

2. Continuous queries:

   (a) When a query has a selection and a join, Niagara explores two options, one to do selection first, followed by multiple joins, and the other to do join first then the selection.

      i. What is the drawback of the second option? ...2

      ii. Suggest how to minimize this disadvantage by pushing a semijoin under the join. ...4

   (b) Suppose you have continuous queries that specify a set of keywords, and should be notified of all new documents that contain the specified set of keywords.

      i. For the special case where queries contain only a single keyword, how would you find queries that are relevant to a given document? ...3

      ii. Now if queries can contains more than one keyword, how can you efficiently find all relevant queries? ...6

3. Snapshot isolation

   (a) Consider two transactions T1 and T2, which demonstrate write skew. Suppose both transactions are now modified so both update the same item x. Can write skew happen in this case? Explain why or why not. ...3

   (b) Suppose you have a transaction using a selection predicate that selects all tuples in r with $r.X = k$ for some specified constant, and a transaction that inserts tuples into r.

      i. Snapshot isolation does not prevent the phantom problem. Explain why, using the above transactions. ...5

      ii. Suppose you also have a relation $x(X)$ with primary key X, and which stores all X values in relation r.
      Suggest how you can ensure that the phantom problem for the above predicate can be prevented under SI by adding appropriately constructed "select for update" queries and inserts to both the transaction doing the selection, and to the transaction performing the insertion. ...7

4. Bigtable/PNuts

   (a) Why does Bigtable use per-server commit logs instead of per-tablet commit logs? Explain briefly. ...3

   (b) When SSTables are merged, what happens to lookups that are ongoing on the SSTables being merged? Do they have to be stopped? And when can old tables be garbage collected? Explain briefly. ...6

   (c) A lock lease period in Chubby can be set short or long. What are the tradeoffs in deciding the lease period? ...3

   (d) In PNUTS, what happens to lookups if a tablet has been split, but the router table was out of date? Explain how the system ensure that the correct answer is sent even in this case. ...5

   (e) The PNUTS mentions that a secondary index is just like a materialized view. Given a relation $r(PK, A, B)$, stored partitioned on attribute PK, explain what materialized view would correspond to a secondary index on B. ...3

5. Probabilistic data:

   (a) Consider the example relation(model,location), in the paper, and explain why count based allocation of a tuple(Tata Indigo, Maharashtra), makes more sense than uniform allocation, if the children of Maharashtra in the hierarchy consist of Mumbai, Pune, Nashik and Jhumritalaiya. ...5

   (b) Explain in your own words, what is the intuition behind the conditions for safety of the projection operator, described in the Dalvi/Suciu paper. ...5

6. Column stores:

   (a) Suppose you model a column representation in a column store logically as a table $ci(tid, value)$. Under what condition can you store the column representation as an array, without storing tid? ...2

   (b) To track which tuples satisfied a particular predicate, one can use either a bitmap or a list of tids. Give conditions under which a bitmap is preferable to a list of tids, and vice-versa. ...3

(c) Now suppose you have a fact table $f(A, B, C)$, where B and C are foreign keys into dimension tables $d(B, S)$ and $e(C, T)$. Fact table f is stored as $fA(tid, A)$, $fb(tid, B)$ and $fc(tid, C)$. Now consider the SQL query

```
SELECT A
FROM f, d, e
WHERE f.B=d.B and f.C=e.C
```

The invisible join can be used for this query. Your task though is to implement the invisible join using relational algebra operations, such as select, join and semijoin, on the broken up fact table and the dimension tables. (Don't worry about physical details such as hash tables etc, the goal is to carry out the same logical steps as invisible join).

...5

Total Marks = 100