School of Information Technology
IIT Kharagpur

Course Id: IT60113 Advanced Database Systems (End Semester Exam)

Date: November 23, 2010 **Total Time: 3 Hours**
Max. Marks: 100

Answer Q. No. 1 and any 3 from the rest. You may answer the questions in any order. However, all parts of the same question must be answered together. Clearly state any assumption that you make.

1. Consider the following set of six 3-D points: (1,0,5), (10,15,15), (10,20,30), (5,5,5), (10,20,20), (15,20,25). (Help: Consider as if the above points represent 3 dimensional histograms)
   (a) Construct an R-Tree with M=2 and m=1 for the above set of data points, assuming that the data arrived in the above sequence. Show each intermediate step for insert. You do not need to write the algorithms, but must clearly (in one sentence each) explain the steps as you proceed.
   (b) Once the R-Tree has been constructed, the entry (10,20,30) is to be deleted. Show the new R-Tree, explaining the individual steps (one sentence each).
   (c) Consider a query point (16,16,20). Find its nearest neighbor using the R-Tree constructed in (a) above (i.e., before deletion in (b)) following MINDIST ordering while doing DFS. Apply suitable pruning strategies using MINDIST and MINMAXDIST based heuristics. Explain each step in brief as you proceed.

[20+5+15=40]

2. Consider the following set of three transactions:

   T1: r(x); r(z); w(x); w(y); w(z)
   T2: r(z); r(y); w(z); w(x)
   T3: r(x); r(y); w(y); w(z)

   For the above three transactions, generate non-trivial (i.e., not serial unless that is the only way):
   (a) Conflict/View serializable schedule using 2PL
   (b) Conflict/View serializable schedule using Timestamp based protocol with Thomas' Write rule

   Clearly identify the corresponding serial schedules and the conditions that determine the order of the equivalent serial schedules.

[10+10=20]

3. Briefly explain the following concepts, giving examples wherever applicable:
   (a) Concurrency control scheme with multiple granularity along with compatibility matrix of its locks
   (b) Handling of coordinator failure and network partitioning during two phase commit protocol in distributed databases
   (c) Four levels of consistency in SQL
   (d) SS-Tree **[5×4=20]**

4.

(a) Consider a database having two tables: EmpSale (EmpName, Sale) and EmpMgr (EmpName, MgrName). (The Sale column of the EmpSale table essentially contains the sum total of sales for an employee and all those who are directly under him). Write suitable ORACLE trigger(s) which will insert/update the sales of an employee's immediate manager (as obtained from EmpMgr table) whenever EmpSale table is updated or rows are inserted in it. You need not handle delete. No penalty for minor syntax errors. If necessary, for the topmost employee, consider that he is his own manager.

(b) Consider the two tables shown below. These are similar to the tables mentioned in 4(a). However, the difference is that the Sale column of EmpSale table contains the direct sale of the corresponding employee and not the sum of sales as mentioned above. Assume that these tables are represented as base predicates in Datalog. Write suitable Datalog derived predicates that will return a Manager name, Name of his reportee and the corresponding Sale amount. There is no need to sum up the sales. It may be noted that the employee manger relation can go up to any level and not only up to the level shown in the example. Write the result of querying your derived predicate on the sample data given below.

[10+10=20]

EmpSale

EmpName	Sale
A	100
B	200
C	300
D	400
E	500

EmpMgr

EmpName	MgrName
A	B
B	C
D	C
C	E
E	E

5. Consider the following three tables T1, T2 and T3 available in sites A, B and C, respectively:

T1

A1	**A2**
A	1
B	2
C	3
D	4
E	5

T2

A1	**B2**
A	a
A	b
B	h
C	e
C	f
D	g
E	h
C	k

T3

B2	**C2**
a	150
a	200
b	120
f	100
e	180
g	149
g	152

In both the questions below, ignore the actual join computation time and only consider the data transfer time. Assume the cost of transferring one cell of data (i.e., one row-column combination of data) for any table from A to B (also B to A) is 10, B to C (also C to B) is 20 and C to A (also A to C) is 15.

(a) We need to compute the natural join between T1, T2 and T3 and the result should be available in site B. Which would be the optimum strategy for computing this natural join?

(b) If we wish to compute natural join between T2 and T3 and make the result available in Site A, determine if using semijoin strategy either at B or C and transferring the result to A would be beneficial compared to transferring both T2 and T3 to A and computing the join there. **[12+8=20]**