Databases and Web Applications (320302)

Final Exam Fall 2011

Logistics

- You have **120 minutes** for the test.
- You can achieve a maximum of **52.5** points, from which only **50** are needed for a perfect score.
- Don't forget to enter your name – we cannot grade if you if it is not present or illegible!
- *Different problems test different skills and knowledge, so do not get stuck on one problem.*

Name:

(To be used for correcting, do not write into box below)

Task	**1.1**	**1.2**	**1.3**	**2.1**	**2.2**	**2.3**	**3.1**	**3.2**	**4.1**	**4.2**	**5.1**	**5.2**	**6.1**	**6.2**	**6.3**	**6.4**	**Total**
Total	**1**	**3**	**5**	**3**	**3**	**3**	**6**	**5**	**4**	**4**	**3**	**4**	**3**	**3**	**1.5**	**1**	**50**
Reached																	

1 Indexing

Consider the B+ tree index shown in the Figure 1. Each intermediate node can hold up to five pointers and four key values. Each leaf can hold up to four records, and leaf nodes are doubly linked as usual, although these links are not shown in the figure. I1 is the root node of the tree.

Task 1.1 (1 pts): Name all the tree nodes that must be fetched to answer the following query: “Get all records with search key greater than 38.”

Task 1.2 (3 pt): Name a search key value such that inserting it into the (original) tree would cause an increase in the height of the tree. Explain why the height will increase.

Task 1.3 (5 pts): Note that subtrees A, B, and C are not fully specified. Nonetheless, what can you infer about the contents and the shape of these subtrees? Consider the height, range of search keys, and the minimum number of key values and pointers that each intermediate node has.

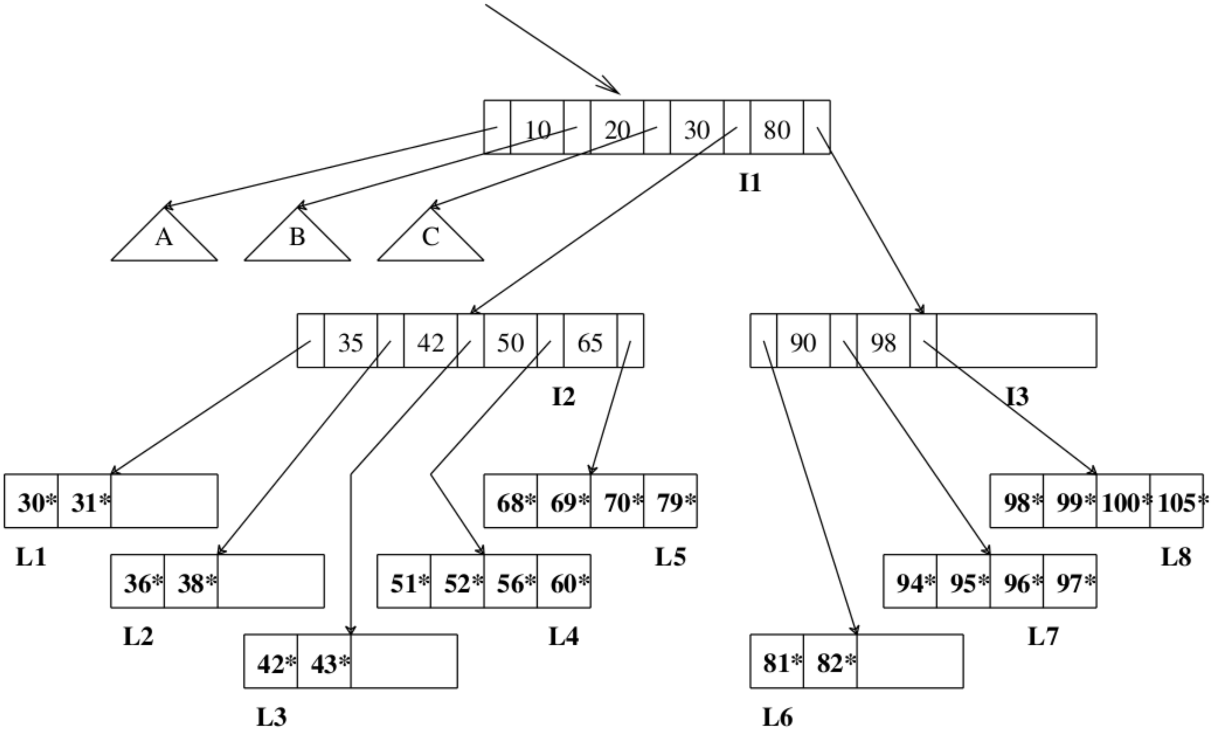

Figure 1

2 Security

Task 2.1 (3 pts): Privileges in Role-Based Access Control (Discretionary Access Control):
(a) Ann creates a table *SECRET1* and executes the following command:

GRANT SELECT, INSERT ON SECRET1 TO Bill.

Sometime later she enters this command:

GRANT DELETE ON SECRET1 TO Bill.

What rights on *SECRET1* does Bill now have?

(b) Ann creates a table *SECRET2* and executes the following command:

GRANT SELECT ON SECRET2 TO BILL WITH GRANT OPTION.

Then Bill executes:

GRANT SELECT ON SECRET2 TO Chris.

Later Ann enters:

REVOKE SELECT ON SECRET2 FROM Bill.

Who will lose the *SELECT* right?

(c) Now Ann creates a table *SECRET3* and executes the following command:

GRANT SELECT ON SECRET3 TO Bill.

Bill executes the command

*CREATE VIEW MY_SECRET AS SELECT * FROM SECRET3*

and then

GRANT SELECT ON MY_SECRET TO CHRIS.

Is it possible to share the secret in this way with Chris?

Task 2.2 (3 pts): Explain role-based access control.

Task 2.3 (3 pts): You are given a database with 2 tables (T1, T2) and 3 users (U1, U2, U3).

Assign security classes to the tables and users, according to the Bell LaPadula model, such that:

- U1 can read from T1 and T2 but can only write in T2.
- U2 can't read from any table but can write in both.
- U3 can read from T1 and can only write in T2.

3 Query Processing

Task 3.1 (3+3 pts): Consider the following database at MusicStoreDotCom:

CUSTOMER (Cid, Cname, Ccity, Cemail)
ALBUM (Album#, Author, Price, Genre, Year)
ORDER (Order#,Cid,Album#, Order_date,Payment_type)

and the following query: "*Customer_email, Album#, Year for customers who ordered albums released after 1991*"

SELECT C.Cemail, A.Album#, A.Year
FROM Customer C, Album A, Order O
WHERE A.Album# = O.Album# AND O.Cid = C.Cid AND A.Year > 1991

a) Draw the logical query tree for the query.

b) Use a heuristics for algebraic query optimization to transform/restructure the query tree you generated at a) above into a more efficient query tree. Explain why your revised query plan is better.

Task 3.2 (2+3 pts): The *index-nested-loop-join(joinatt,T1,T2)* for a join attribute "joinatt" appearing in table T1 and T2 can help to evaluate a join predicate substantially faster than a naive *nested-loop-join(joinatt,T1,T2)*. Compare efficiency of both algorithms in detail, based on pseudo code for both, and derive a rule which, based on the ratio of join-matching tuples in T1 and T2, determines whether "T1 join T2" or "T2 join T1" will be more efficient.

4 Transactions

Task 4.1 (4 points): Name and briefly explain the four core properties the transaction concept offers.

Task 4.2 (2+2 points): Consider a database with objects *X* and *Y* and assume that there are two transactions *T*1 and *T*2. Transaction *T*1 reads objects *X* and *Y* and then writes object *X*. Transaction *T*2 reads objects *X* and *Y* and then writes objects *X* and *Y*.

a) Give an example schedule with actions of transactions *T*1 and *T*2 on objects *X* and *Y* that results in a write-write conflict.

b) Show that *Strict 2PL* disallows the above schedules.

a.

b.

c.

5 Normal Forms

Task 5.1 (1+2 points):

a) What does the functional dependency X Y mean?
b) Define the concept of a “key” formally, based on the concept of functional dependencies.

b.

-

Task 5.2 (4 pts): Suppose relation S = ABC has attributes A, B, and C. Attribute A uniquely determines B, and C uniquely determines A. Let S be decomposed into S1 = AB and S2 = AC.

a) What are the candidate keys of S, S1, and S2, respectively?
b) Does the decomposition remove redundancy? If so, why?
c) Does the decomposition lead to tables that are overall smaller in number of tuples? If so, why?
d) Does the decomposition lead to tables for which it is easier to write query operations? If so, why?

6 Web Services

Task 6.1 (3 points): Briefly describe SOAP, mentioning its advantages and disadvantages.

Task 6.2 (3 points): Briefly describe REST, mentioning its advantages and disadvantages.

Task 6.3 (1.5 bonus points): Describe one kind of service where SOAP is more appropriate, and one where REST is more appropriate.

Task 6.4 (1 bonus point): You have a website where users can listen to music. You want to offer external developers a service where they can analyze statistics about each user (e.g. what are his favorite songs, what songs he has recently listened to and so on). Would you use SOAP or REST to expose your service? Why?

--end of exam--