B561 Final Exam

December 17, 2004

C. M. Wyss

Instructions:

You have 2 hours to complete this exam. This exam has 4 questions (5 pages total). The exam is worth 100 points. Point counts are indicated for each question. Make sure you allot your time so you can answer all questions.

Question 1
(30 points total)

Suppose we have three relations, R(A, B, C, D, E), S(C, D, F, G), and T(A, B, H, I), with the following characteristics:

- R has 10,000 tuples.
- S has 1,000 tuples.
- T has 100 tuples.
- Each field (A through I) is 10 Bytes large.
- Each field (A through I) contains numbers between 0 and 10 (inclusive).
- The values in each field (A through I) are uniformly distributed.
- T is stored sorted on H, but there are no indices for it.
- There are hash indexes on both R(A, B) and R(C, D) (assume 1.2 accesses to find the address of a record given a search key).
- There are both a B+-tree on S.F (of height 2) as well as a hash index on S.F (assume 1.2 accesses to find the address of a record given a search key).

Consider the following SQL query. Two proposed logical plans for this query are shown in Figure 1 (below).

```
SELECT R.E
FROM   R, S, T
WHERE  R.A = T.A AND R.B = T.B AND
       R.C = S.C AND R.D = S.D AND
       S.F >= 3 AND S.F <= 6 AND
       T.H = 0;
```

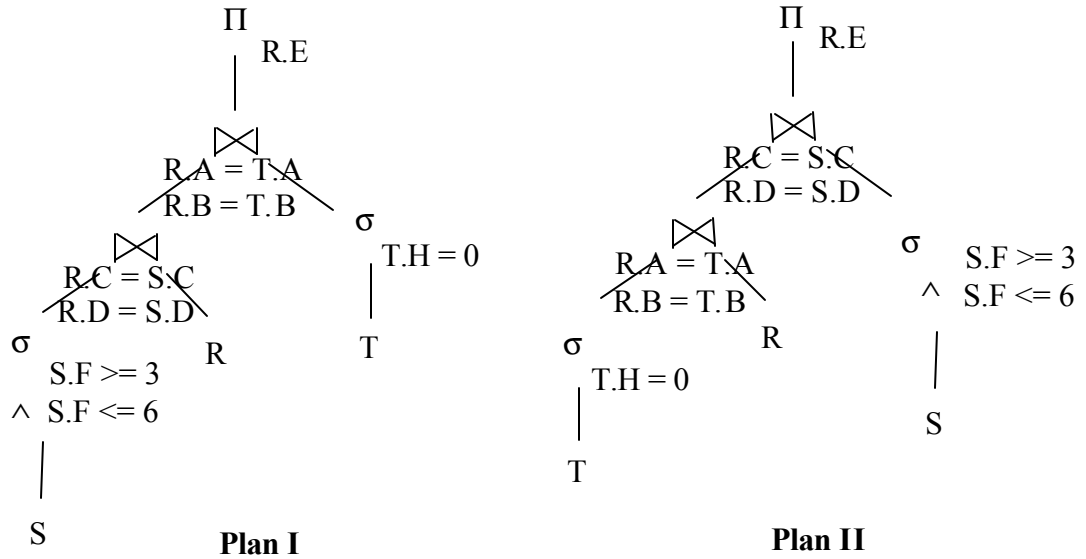

Figure 1: Two proposed logical plans.

Answer questions (a) through (d) (below) based on this information.

a) (**10 points**) Prove that both Plan I and Plan II are equivalent to the original query.
b) (**5 points**) Provide annotated versions of both plans that give execution strategies for the given query. At each node in your annotated plan, justify your choice of algorithm (e.g. Index Nested Loop join vs. Block Nested Loop join).
c) (**10 points**) Give the cost (in block I/Os) for each of your annotated plans from part (b). Assume the block size is 1 KB and each process has 12 blocks of main memory allocated to it. Justify explicitly any assumptions you make that are not in the question statement.
d) (**5 points**) Which plan is better?

Question 2
(30 points total)

Suppose our database has five pieces of information (call them A, B, C, D, E). The initial values before time t_0 are A=0, B=1, C=2, D=3, E=4. Consider the transaction schedule shown in Figure 2 (below). In the schedule, a term like "W(x, y)" means that a transaction writes value y into data slot x.

Time	T_1	T_2	T_3	T_4
t_0	R(A)			
t_1		R(A)		
t_2	W(A,1)			
t_3			R(B)	
t_4				W(B,4)
t_5			W(B,3)	
t_6		W(A,2)		
t_7	R(A)			
t_8		W(A,2)	W(B,3)	
t_9				W(A,4)
t_{10}		W(A,2)		

Figure 2: Schedule of transactions.

Given this, answer questions (a) through (f), below.

a) (**5 points**) Is the schedule conflict serializable? Prove your answer. If it is conflict serializable, give an equivalent serial order for the transactions.
b) (**5 points**) Can you insert data locks so as to make the schedule strict 2PL? If so, show the resulting schedule. If not, why not?
c) (**5 points**) Can you insert data locks so as to make the schedule non-strict 2PL? If so, show the resulting schedule. If not, why not?
d) (**5 points**) Exhibit a commit order that makes the schedule recoverable.
e) (**5 points**) What is the value of data slot A in main memory at time t_7? Will this value necessarily be reflected in the stable database? If not, why not?
f) (**5 points**) Suppose transaction T_1 commits at time t_{11} and transaction T_3 commits at time t_{12}, and there is a crash immediately afterward. What should the data values contain in the stable database after recovery? Will this be in line with the ACID guarantees? If not, why not?

Question 3
(40 points total)

Write an essay of between 2 and 10 double spaced blue book pages on the following topic. Some point form is okay, as long as the flow of your ideas is clear.

> What are the three levels crucial to our conception of a database? Give the three levels, and give examples of abstractions appropriate to each. How does the notion of *data independence* work with respect to the three levels?
>
> With respect to these three levels, where does query processing and optimization take place? Does query processing and optimization perfectly respect the principle of data independence? Explain your answer in detail. Including a technical example will probably help your explanation.

Bonus Question
(2 points total)

Which of the following was *not* offered at the mini-database workshop on the last day of class?

a) coffee
b) donuts
c) juice
d) Christmas cookies