Name _____ Class Account _____

UNIVERSITY OF CALIFORNIA
Department of EECS, Computer Science Division

CS186 Hellerstein

Fall 2007 Final Exam

Final Exam: Introduction to Database Systems

This exam has six problems, worth different amounts of points each. Each problem is made up of multiple questions. You should read through the exam quickly and plan your time management accordingly. Before beginning to answer a problem, be sure to read it carefully and to *answer all parts of every problem!*

You **must** write your answers on the exam. Extra answer space has been provided at the back in case you run out of space while answering. If you run out of space, be sure to make a "forward reference" to the page number where your answer continues. ***Do not tear pages off of your exam!***

Good luck!

Name _____ Class Account _____

Question 1: Selectivity Estimation [12 points]

Consider a database of two relations **X(**a, b, c**)** and **Y(**c, d, e**)**, with statistics as shown to the right. The following SQL query is to be executed:

```
SELECT b, d
  FROM X, Y
 WHERE X.c = Y.c
   AND X.b = 7
   AND Y.d = 5
```

Relation X	20,000 tuples
index on X.a	NKeys(X.a) = 5
index on X.b	NKeys(X.b) = 4
index on X.c	NKeys(X.c) = 10
Relation Y	60,000 tuples
index on Y.c	NKeys(Y.c) = 4
index on Y.d	NKeys(Y.d) = 3
index on Y.e	NKeys(Y.e) = 6

a. [6 points] Given plan $P_1 = \pi_{b,d}\,[\sigma_{X.b=7\ \&\ Y.d=5}(X \bowtie Y)]$ estimate the cardinality of each of the following:

i. Cardinality of output of $X \bowtie Y$:

ii. Cardinality of output of $\sigma_{X.b=7\ \&\ Y.d=5}\ (X \bowtie Y)$:

iii. Cardinality of output of $\pi_{b,d}\,[\sigma_{X.b=7\ \&\ Y.d=5}(X \bowtie Y)]$:

b. [6 points] Given plan $P_2 = \pi_{b,d}\,[(\sigma_{X.b=7}\ X) \bowtie (\sigma_{Y.d=5}\ Y)]$, estimate the cardinality of after each operation as follows:

i. Cardinality of output of $\sigma_{X.b=7}\ X$:

ii. Cardinality of output of $\sigma_{Y.d=5}\ Y$:

iii. Cardinality of output of $\pi_{b,d}\,[(\sigma_{X.b=7}\ X) \bowtie (\sigma_{Y.d=5}\ Y)]$:

Question 2: Join Costs [6 points]

In this question, you will consider joining two relations from an e-commerce database. The CARTS relation represents shopping carts of items. The CONTENTS relations has the contents of all the carts, with a *foreign key* called cartID to CARTS. cartID is specified with a NOT NULL constraint.

CARTS(cartID, customerID, date, comment**)**
CONTENTS(cartID, productID, quantity, price**)**

Assume:

- Fixed size tuples in both relations
- 10 CARTS tuples per page
- 100 CONTENTS tuples per page
- 1000 pages in the CARTS relation
- 5000 pages in the CONTENTS relation
- Join on CARTS.cartID=CONTENTS.cardID

a. [2 points] How many I/O requests are required for a page-oriented nested loops join, with CARTS as the outer relation and CONTENTS as the inner relation? Assume that every page reference generates an I/O (i.e. absolutely no hits in the buffer pool).

b. [2 points] Now, again using CARTS as the outer relation and CONTENTS as the inner relation, assume that the buffer manager is used properly, and:

- There are 1002 frames in the buffer pool
- The buffer manager starts empty
- One frame is pinned by the join for the purpose of holding output tuples until they are ready to be flushed to disk. This frame is not unpinned by the join until the end of the query.
- One buffer frame is pinned by the join and holds the current page of the outer relation at all times. All I/Os to the outer relation are placed explicitly into this frame, which is not unpinned until the end of the query
- The buffer manager is running LRU replacement policy
- No other queries are running in the system.

How many I/Os are required for a page-oriented nested loops join?

c. [2 points] How many I/Os are required for a block (“batch”) nested loops join, with CONTENTS as the outer relation, CARTS as inner relation, and 1000 pages per block (“batch”)? Assume that 1000 buffer frames are pinned throughout, and hold the current outer “batch” at any time.

Name _____ Class Account _____

Question 3: Concurrency Control [17 points]

a. Consider the following transaction schedule

XACTID	Time 1	Time 2	Time 3	Time 4	Time 5	Time 6	Time 7	Time 8	Time 9
T1	R(X)	R(Y)							W(X)
T2					R(Z)	W(X)	R(Y)		
T3			R(Z)	W(Y)				R(Z)	

i. [2 points] Draw the schedule's dependency graph.

ii. [2 points] Is this schedule *view equivalent* to a serial schedule that has T3 preceding T2, which in turn precedes T1? (YES / NO) Please explain in 2 lines or less.

iii. [2 points] Is the schedule *conflict serializable*? (YES / NO) Please explain in 2 lines or less.

b. Consider the following transactions schedule

XACTID	Time 1	Time 2	Time 3	Time 4	Time 5	Time 6	Time 7	Time 8	Time 9
T1	R(X)		W(K)	W(X)			R(K)		Abort
T2		R(Y)			R(X)	W(Z)		Commit	

a) [1 point] Can this schedule be produced by 2PL? (YES / NO)

b) [1 point] Can this schedule occur under a Strict 2PL locking policy? (YES / NO)

c) [2 points] How would you change this schedule to produce a cascading abort scenario? Do not modify the above schedule in more than 3 places.

c. [7 points] Circle TRUE or FALSE to reflect the validity of the following statements

i. For any schedule S1 and any serial schedule S2, if S1 is conflict equivalent to S2, then S1 is conflict serializable.

TRUE / FALSE

ii. For any schedules S1 and S2, if S1 and S2 are conflict serializable, then S1 and S2 are conflict equivalent.

TRUE / FALSE

iii. All serial schedules avoid cascading aborts.

TRUE / FALSE

iv. Lock upgrades can cause deadlock.

TRUE / FALSE

v. In optimistic concurrency control, read-only transactions do not need a validation phase.

TRUE / FALSE

vi. A SIX lock is compatible with an IS and IX lock.

TRUE / FALSE

vii. Only Strict 2-phase locking ensures conflict serializable schedules.

TRUE / FALSE

Name _____ Class Account _____

Question 4: Recovery [21 points]

a. Consider the following sequence of actions taken by transactions T1, T2 and T3, against a DBMS with pages P1, P2, P3, P4 and P5.

i. [4 points] Below is the corresponding database log, containing all operations since the database was first initialized. Fill in the 7 missing entries shown in gray.

LSN	PrevLSN	UndoNext LSN	XactID	Type	pageID	Before	After
10	null	--	T1	update	P1	17	22
20	--	--	--	begin_chk	--	--	--
30		--	T2	update	P3	31	42
40	--	--	--	end_chk	--	--	--
50	null	--	T3	update	P5	14	9
60	10	--	T1	update	P2	12	13
70	60	--	T1	update	P1	22	17
80	70	--	T1	commit	--	--	--
85	80	--	T1	end	--	--	--
90	30	--	T2	update	P4	3	6
100	--	--	--	begin_chk	--	--	--
110		--	T3	abort	--	--	--
120	90	--	T2	update	P4	6	5
130			T3	CLR	P5		
140	120	--	T2	update	P5		100

ii. [1 point] At which LSN will the ANALYSIS phase start?

iii. [4 points] Show the Transaction Table and Dirty Page Table as it would look at the end of the ANALYSIS phase

Xact Table

Dirty Page Table

iv. [1 point] At which LSN will the REDO phase start?

v. [5 points] Fill in the table containing for each page, its PageLSN and its value, after the end of the REDO phase.

	P1	P2	P3	P4	P5
PageLSN					
Value					

vi. [1 point] At which LSN will the UNDO phase start?

vii. [1 point] What LSNs does the "ToUndo" set contain in the beginning of the UNDO phase?

viii. [4 points] Fill in the log with the CLR information that the UNDO phase will generate.

LSN	PrevLSN	XactID	Type	pageID	Before	After

Question 5: Logical Database Design [20 points]

A	B	C	D
1	1	1	2
1	2	1	3
2	2	1	4
2	1	1	5

a. [8 points] Consider the relation **R(**A, B, C, D**)** with the *consistent* instance above. A user tries the following query:

```
INSERT INTO R VALUES (2, 2, 2, 2);
```

The user's transaction is aborted due to a consistency violation. Consider each of the following integrity constraints *independently* and circle it if it could have caused that violation:

i. (A, B, C) is a candidate key of R

ii. (A, B) is the primary key of R

iii. D → B

iv. C is a Foreign Key to relation S

b. [2 points] Ignoring the constraints in part (a), is (CD) a superkey of relation R above? Justify your answer.

c. [2 points] What is the maximal superkey of relation R (i.e. the superkey with the most attributes)?

d. You are told that the Functional Dependencies on R are F = {AB → CD, D → B}.

   i. [2 points] Does AB → CD violate Third Normal Form? Justify your answer.

   ii. [2 points] Does D → B violate Third Normal Form? Justify your answer.

   iii. [2 points] Write a decomposed schema for R that is in Boyce-Codd Normal Form.

e. [2 points] Consider the following decomposition:

   F = {AB → CDE, BE → X, A → E}

   R1(ABC), R2(BCDEX)

   Is this a lossless decomposition? You can consider the opinions of two self-styled experts if you like:

   i. Prof. Orange asserts: “This is a lossless join decomposition. By repeated application of Armstrong’s axiom of transitivity we can show that (ABC) is a superkey of the original relation R. As a result, we are guaranteed that R1 has no duplicates, and the join of R2 and R1 will be identical to the original relation R.”

   ii. Prof. Green says: “On the contrary, the decomposition is not lossless, it is lossy. Consider the intersection of the attributes of R1 and R2, namely (BC). If you form the attribute closure of BC with respect to F, you will find that you get neither ABC nor BCDEX, and hence the join of R1 and R2 may well be a superset of R.

Name _____ Class Account _____

Question 6: SQL [8 points]

The Bureau of Lollipop Statistics has hired you to build an application to analyze their data. They have a sales history table, S, with the schema **S(**year, month, day, color, profits**)**.

a) [2 points] Find the overall minimum, average, maximum, and standard deviation of the profits column. (Note: the SQL aggregation function for standard deviation is STDDEV.)

```
SELECT _____
  FROM S

_____
```

b) [2 points] Find the average and standard deviation of lollipop profits per day in the month of February, 2007, grouped by color.

```
SELECT _____
  FROM S

_____


_____
```

c) [2 points] Return the year, month, day and profits from yellow lollipops for the highest-profit day ever for yellow lollipops. (If there is a tie, we will accept queries that return one or more of the tied rows.)

```
SELECT year, month, day, profits FROM S _____

_____


_____
```

d) [2 points] Return the color, average and standard deviation of lollipop profits per day for colors whose total profit over time is less than 1,000,000.

```
SELECT * FROM S _____
```

_____

_____

_____

Name _____ Class Account _____

SCRATCH.

Name _____ Class Account _____

SCRATCH