Name:____

CSE 444, Winter 2011, Midterm Examination
9 February 2011

Rules:

- Open books and open notes.
- No laptops or other mobile devices.
- Please write clearly.
- Relax! You are here to learn.
- An extra page is provided in case you run out of space, but make sure to put a forward reference.

Question	Max	Grade
1	30	
2	30	
3	18	
4	22	
Total	100	

Initials:____________________________________

1. (**30** points) **SQL**

   For a given quarter, the table `Student` lists all students registered at the University, the table `Course` lists all courses offered, and the table `Enrollment` encodes the students that attend any course in the current quarter.

```
Student ( sid, sname )
Course ( cid, cname )
Enrollment ( cid, sid )
```

   `Enrollement.sid` is a foreign key that references `Student.sid`.

   `Enrollement.cid` is a foreign key that references `Course.cid`.

   (a) (**8** points) Assume that no two courses have the same name. Call a course "big" if the number of students enrolled is at least 50. Write a SQL query that returns the names of all big courses in decreasing order of their enrollment size.

Initials:____

(b) (**8** points) A “classmate” of a student is <u>another</u> student who is enrolled in at least one same class. Write a SQL query that returns all pairs of classmates. You should return their student IDs, as their names do not uniquely identify them.

Initials:____

(c) (**14** points) Write a SQL query that computes for each student id in the `Student` table the total number of different classmates that this student has across all courses. Note that a student may be so busy with research (or parties ⌣̈) that he or she may choose not to enroll to any course that quarter, and hence has no classmates. Such students should still appear in the result, with 0 as the number of their classmates.

Hint: this is not required, but it may help you to make use of an earlier query.

Initials:____

2. (**30** points) **Conceptual Design**

   (a) (**12** points) A friend is impressed by your skills in CSE 444, and asks you to help her design a database to model doctors' offices across Seattle. She started the E/R diagram below which already contains all the entity sets, their attributes and relationships she wants you to consider. You need to finish the diagram and define a number of constraints to ensure that you model the semantics of an office as closely as possible.

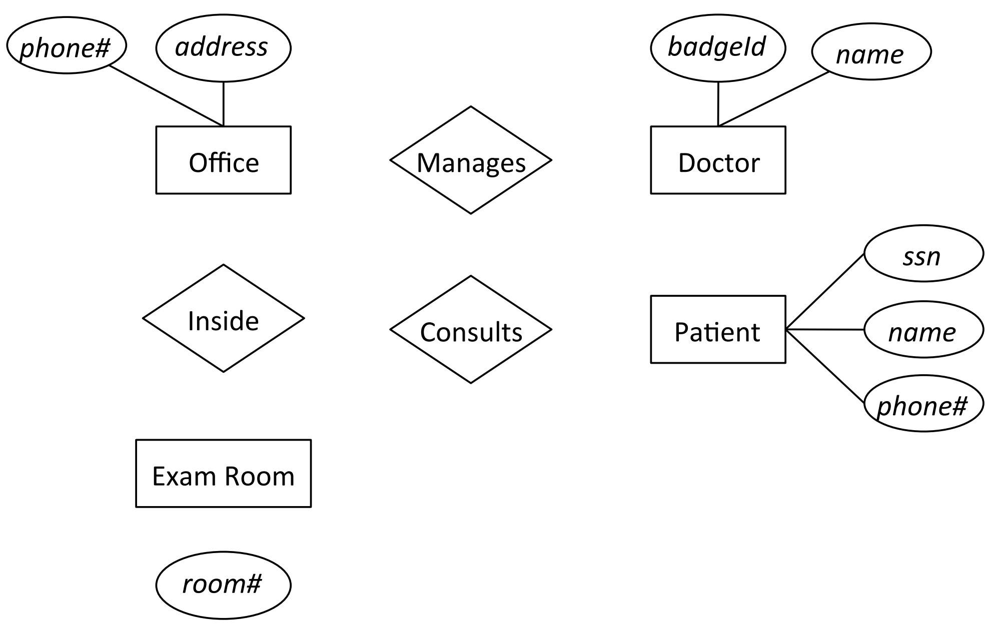
   Make sure that you do not impose additional constraints not defined by the model.

   i. (**4** points) An office may be managed by at most one doctor. A doctor is uniquely identified by their badgeId, and may manage more than one office. Draw the necessary constraints to capture this requirement.

   ii. (**5** points) An office is identified by its address, and contains one or more exam rooms. A room can be identified by its room number, and the office that it is in. Capture these constraints by drawing on or modifying the diagram where necessary.

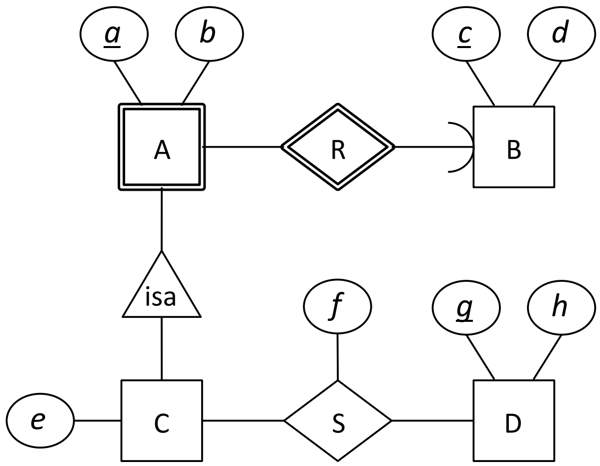
   iii. (**3** points) When a patient visits an office, he or she has a consultation with a doctor in an examination room. Each patient is uniquely identified by their SSN.

   Draw your answer directly onto the figure below. Make sure that any arrows, lines, double lines etc, are clearly visible.



Initials:____

(b) (**12** points) Convert the E/R diagram below to a database schema. Indicate the keys for each table in your answer. You do not need to write SQL DDL commands.



Initials:____

(c) (**6** points) Suppose we are told that `R(A, B, C, D)` is in BCNF, and that 3 out of the 4 FDs listed below hold for R. Choose the FD that R does <u>not</u> satisfy, and explain your reasoning.

$$\begin{aligned} 1 &: A \rightarrow BCD \\ 2 &: BC \rightarrow A \\ 3 &: CD \rightarrow B \\ 4 &: D \rightarrow C \end{aligned}$$

Initials:____

3. (**18** points) **Logging and Recovery**

Your database server crashed due to a power outage. After rebooting, you find the following log and checkpoint information on disk, and begin the recovery process. We assume that STEAL/NO FORCE policy is used, and we use the ARIES method for recovery.

LSN	**Record**	**prevLSN**	**undoNextLSN**
30	update: T3 writes P5	null	-
40	update: T4 writes P1	null	-
50	update: T4 writes P5	40	-
60	update: T2 writes P5	null	-
70	update: T1 writes P2	null	-
80	Begin Checkpoint	-	-
90	update: T1 writes P3	70	-
100	End Checkpoint	-	-
110	update: T2 writes P3	60	-
120	T2 commit	110	-
130	update: T4 writes P1	50	-
140	T2 end	120	-
150	T4 abort	130	-
160	update: T5 writes P2	Null	-
180	CLR: undo T4 LSN 130	150	50

Transaction Table at time of checkpoint

Transaction ID	**lastLSN**	**Status**
T1	70	Running
T2	60	Running
T3	30	Running
T4	50	Running

Dirty Page Table at checkpoint

Page ID	**recLSN**
P5	50
P1	40

(a) (**2** points) The log record at LSN 60 denotes an update to page P5 by transaction T2. Was this update to page P5 successfully written to disk? If yes, at what point could that have happened?

(b) (**2** points) The log record at LSN 70 denotes an update to page P2 by transaction T1. Was this update successfully written to disk? If yes, at what point? Briefly explain your answers.

Initials:____

(c) (**6** points) At the end of the Analysis phase, what will the Transaction and Dirty Page Tables look like? Populate your answers in the tables below.

Transaction ID	**lastLSN**	**Status**

Page ID	**recLSN**

(d) (**6** points) At which LSN in the log will REDO begin? Which log records will be redone? List their LSNs, and briefly justify your answers.

Initials:____

(e) (**2** points) For each of the following questions circle the right answers.

i. In an UNDO only logging scheme, what buffer management policies apply? (circle one of STEAL or NO STEAL, <u>and</u> one of FORCE or NO FORCE)

STEAL / NO STEAL FORCE / NO FORCE

ii. During recovery with REDO only logging, we cannot use a checkpoint for which we see a <START CKPT($\ldots T_i \ldots$)> record but no corresponding <END CKPT> record.

TRUE / FALSE

Initials:____

4. (**22** points) **Concurrency Control**

In the schedules given below, the label $R_i(X)$ indicates a read of element X by transaction T_i, and $W_i(X)$ indicates a write of element X by transaction T_i.

(a) (**4** points) Draw the precedence graph for schedule 1. Is schedule 1 conflict-serializable? If so, what order of the three transactions defines a conflict-equivalent serial schedule?

Schedule 1

$R_2(A)\ \ R_1(C)\ \ R_2(B)\ \ W_2(B)\ \ R_3(B)\ \ R_1(A)\ \ R_3(C)\ \ W_3(C)\ \ W_1(A)$

(b) (**4** points) Draw the precedence graph for schedule 2. Is schedule 2 conflict-serializable? If so, what order of the three transactions defines a conflict-equivalent serial schedule?

Schedule 2

$R_2(A)\ \ R_1(C)\ \ R_2(B)\ \ R_3(B)\ \ W_2(B)\ \ R_1(A)\ \ R_3(C)\ \ W_3(C)\ \ W_1(A)$

Initials:____

(c) (**9** points) Can the two schedules from (a) and (b) occur under (non-strict) 2 Phase Locking? If yes, then add the proper lock/unlock actions to the corresponding schedule(s) in a way compliant with (non-strict) 2PL. Use $L_i(X)$ to denote transaction i locking element X, and $U_i(X)$ to denote transaction i releasing the lock on X. Assume only exclusive locks are used, and make sure that no locks remain held at the end of the schedule.

The schedules below are the same ones as in the previous page, and repeated here for convenience with dotted spaces in-between the actions. You can use this dotted space to add the proper lock and unlock actions, or if you prefer you can rewrite the schedules including the locking actions in the empty space further down the page.

If you believe a schedule cannot occur with 2PL, give a brief explanation.

Schedule 1

$\ldots\ldots\ldots R_2(A)\ldots\ldots\ldots R_1(C)\ldots\ldots\ldots R_2(B)\ldots\ldots\ldots W_2(B)\ldots\ldots\ldots R_3(B)\ldots\ldots\ldots$

$\ldots\ldots\ldots R_1(A)\ldots\ldots\ldots R_3(C)\ldots\ldots\ldots W_3(C)\ldots\ldots\ldots W_1(A)\ldots\ldots\ldots$

Schedule 2

$\ldots\ldots\ldots R_2(A)\ldots\ldots\ldots R_1(C)\ldots\ldots\ldots R_2(B)\ldots\ldots\ldots R_3(B)\ldots\ldots\ldots W_2(B)\ldots\ldots\ldots$

$\ldots\ldots\ldots R_1(A)\ldots\ldots\ldots R_3(C)\ldots\ldots\ldots W_3(C)\ldots\ldots\ldots W_1(A)\ldots\ldots\ldots$

Initials:____________________________________

(d) (**5** points) Circle TRUE or FALSE to reflect the validity of the following statements.

i. For any schedules S_1 and S_2, if S_1 and S_2 are conflict serializable, then S_1 and S_2 are conflict equivalent.[1]

TRUE / FALSE

ii. A SIX lock is compatible with IS and IX locks, i.e. if a transaction holds a SIX lock on an object, then another transaction can take an IS or IX lock on the same object.

TRUE / FALSE

iii. An IX lock is compatible with an IS lock, i.e. if a transaction holds an IS lock on an object, then another transaction can take an IX lock on the same object.

TRUE / FALSE

iv. Strict 2PL prevents deadlocks.

TRUE / FALSE

v. In timestamp-based concurrency control, if a transaction gets aborted, it will be restarted with a <u>new</u> timestamp.

TRUE / FALSE

[1] Two schedules are conflict equivalent, if every pair of conflicting actions appears in the same order in both schedules.

Initials:____

EXTRA PAGE