Instructions:

1. This is a closed book examination. No additional materials are allowed.
2. Exam duration is **2.5 hours** (07:30 PM - 10:00 PM).
3. Answer all the questions.
4. Answer each question in the space provided.
5. You can use the back of the sheets for rough work.
6. The exam consists of 7 questions and 20 pages; make sure you have all of the pages.

Question:	1	2	3	4	5	6	7	Total
Points:	18	16	15	20	15	8	8	100
Score:								

Q1. (18 points)

Answer the following questions with no more than five sentences (no longer than the allocated space).

(a) (3 points) Explain when is PAX block layout better than NSM?

(b) (3 points) Explain what is deletion anomaly.

(c) (3 points) Given relation `R(A, B, C, D, E)` where `AB` is the key, and functional dependencies `AB`→ `CDE` and `B`→`D`, is `R` in Boyce-Codd Normal Form (BCNF)? Explain your answer.

(d) (3 points) Explain what is the ROLLBACK WORK command.

(e) (3 points) Describe one advantage of B+tree index over ISAM.

(f) (3 points) What are ACID properties of transactions. Explain each by ***one*** sentence.

Q2. (16 points)

Consider the following SQL data definition for maintaining information about employees at a hypothetical company.

```
CREATE TABLE emp
   (  num       INTEGER NOT NULL,
      name      VARCHAR(20) NOT NULL,
      dept      VARCHAR(20) NOT NULL,
      salary    INTEGER NOT NULL,
      boss      INTEGER NOT NULL,
      PRIMARY KEY (num),
      FOREIGN KEY (boss) REFERENCES emp (num) );
```

You can assume that there is one president that has herself/himself as the boss, that all other employees have a boss that is someone else and that there are no cycles in the boss hierarchy for anyone other than the president. (A cycle would exist if, for example, Fred was the boss of Mary and Mary was in turn the boss of Fred.) Translate each of the following queries on this schema to SQL or Relational Algebra.

(a) (4 points) The number and name of each employee, excluding the president, together with the number and name of the employee's boss. The result should be sorted by the name of the boss and then by the name of the employee. (SQL)

(b) (4 points) The names of the departments with the highest average salary of their employees. (SQL)

(c) (4 points) The numbers and names of employees who have the president as their boss, and that have a salary among the lowest of those employees who do not have the president as their boss. (SQL)

(d) (4 points) The numbers, names and departments of employees that are not the boss of any other employees. (Relational Algebra)

Q3. (15 points)

Consider the ER model given in Figure 1. This model represents the operations of a pharmacy chain. Please answer the following questions regarding this model.

(a) (3 points) Can a pharmaceutical company have multiple phone numbers? If not, what do you need to do to allow this?

(b) (3 points) If we delete from the database the pharmaceutical company that manufactures a drug, what happens to the drugs that the company manufactures? Justify (in one or two sentences only) your argument.

(c) (3 points) Similar to part (b), but instead of deleting the pharmaceutical company, what if we delete the pharmacy that sells the drug. Do we have to delete the drug too? Why or why not?

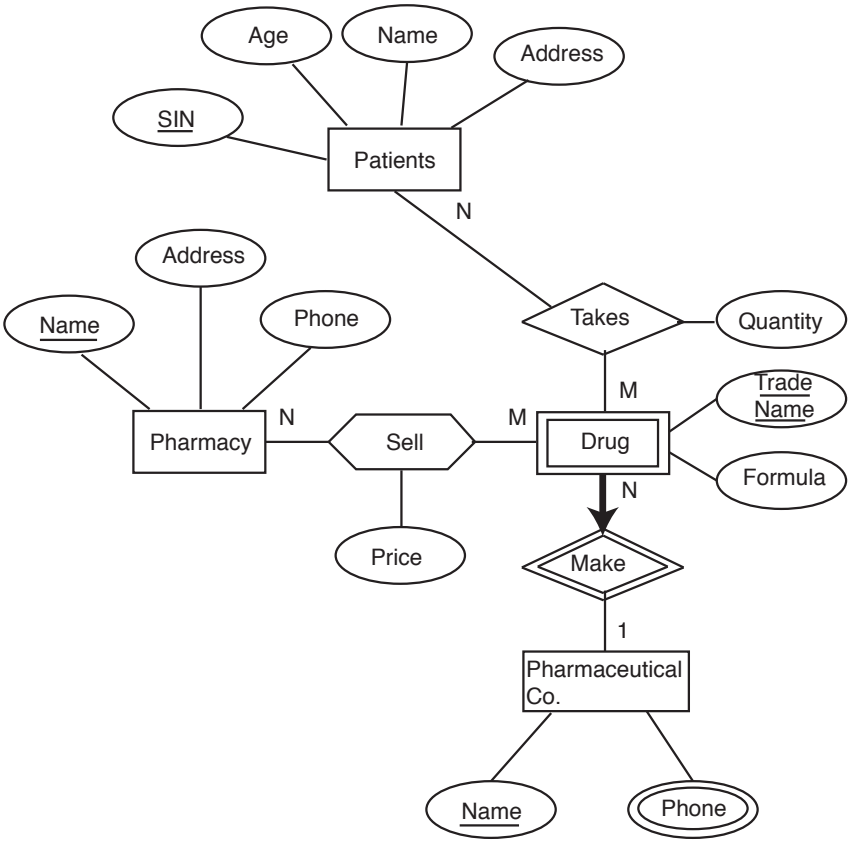

Figure 1: Figure for Question 3

(d) (6 points) Modify the model (by adding to Figure 1, ***not*** by drawing another figure) so that you can represent the following

- Each patient has to have one and only one primary physician, who can be identified by SIN. Each physician has at least one patient. In addition, we want to record the specialty and the date of entry into the profession of each physician.
- Instead of modeling only the fact that a patient takes certain drugs, model the fact that a patient takes drugs that are prescribed by a physician and the prescription date.
- Pharmaceutical companies have long-term contracts with pharmacies. A pharmaceutical company can contract with several pharmacies, and a pharmacy can contract with several pharmaceutical companies. For each contract we want to store a start date, an end date.

Q4. (20 points)

Given relation `R(A, B, C, D)` and the set of functional dependencies `F = {AB→C, C→D, D→A}`.

(a) (5 points) What are all the nontrival FD's that follow from the given FD's? You should restrict yourself to FD's with single attributes on the right side.

(b) (5 points) What are all the keys of `R`?

(c) (5 points) Please state if `R` is in 3NF and explain.

(d) (5 points) Decompose `R` into BCNF.

Q5. (15 points)

Consider a schema of 4 relations, `W(a,b)`, `X(b,c)`, `Y(c,d)`, and `Z(d,e)`. The table size for these 4 relations are 100, 200, 300, and 400 respectively. The number of distinct values for a column `a` in the relation `W` is denoted by $V(W, a)$. Given that $V(W, a) = 20, V(W, b) = 60, V(X, b) = 50, V(X, c) = 100$, $V(Y, c) = 50$, $V(Y, d) = 50$, $V(Z, d) = 40$, $V(Z, e) = 100$.

(a) (3 points) Estimate the sizes of relations that are the results of $\sigma_{c=20}(Y) \bowtie Z$.

(b) (4 points) Estimate the sizes of relations that are the results of $W \bowtie X \bowtie Y$.

(c) (8 points) Suggest index structures that can speed up the two queries in (a) and (b) and explain their corresponding physical plan and their IO cost.

Q6. (8 points)

Consider histories H_1 and H_2 given below:

$$H_1 = r_1(x), r_2(z), r_1(z), r_3(x), r_3(y), w_1(x), w_3(y), r_2(y), w_2(z), w_2(y)$$
$$H_2 = r_1(x), r_2(z), r_3(x), r_1(z), r_2(y), r_3(y), w_1(x), w_2(z), w_3(y), w_2(y)$$

These histories are generated by the following transactions:

$$T_1 = r_1(x), r_1(z), w_1(x)$$
$$T_2 = r_2(z), r_2(y), w_2(z), w_2(y)$$
$$T_3 = r_3(x), r_3(y), w_3(y)$$

(a) (4 points) Draw the serialization graph for H_1 and state whether or not it is serializable. If it is serializable, give the equivalent serial history.

(b) (4 points) Draw the serialization graph for H_2 and state whether or not it is serializable. If it is serializable, give the equivalent serial history.

Q7. (8 points)

For each of the following isolation levels, give an example of an execution history that respects the specified level of isolation, but is not seriablizable.

For example, for "Read uncommitted", we can have an execution history over two transactions

T1:	T2:
UPDATE User	
SET pop = 0.99	
WHERE uid = 142;	SELECT AVG(pop)
	FROM User;
ROLLBACK;	
	COMMIT;

(a) (4 points) Read committed

(b) (4 points) Repeatable read

Extra Notes Page 1

This page will not be marked unless otherwise specified beside a specific question in the exam package.

Extra Notes Page 2

This page will not be marked unless otherwise specified beside a specific question in the exam package.

Extra Notes Page 3

This page will not be marked unless otherwise specified beside a specific question in the exam package.