MCS243 Final Exam

Serial #:

This exam is closed-book and mostly closed-notes. You may, however, use a single 8 1/2 by 11 sheet of paper with hand-written notes for reference. (Both sides of the sheet are OK.)

Please write your name only on this page. Do not turn the page until instructed, in order that everyone may have the same time. Then, be sure to look at all problems before deciding which one to do first. Some problems are easier than others, so plan your time accordingly. You have 90 minutes to work.

Write the answer to each problem on the page on which that problem appears. You may use the back of that same sheet if you need additional space. You may also request additional paper, which should be labeled with your test number and the problem number.

To get partial credit for a problem that you answer wrong, you will need to explain your reasoning.

If you are stuck, I am willing to sell you a hint for some points.

Name: ____

Problem	Page	Possible	Score
1	2	12	
2	3	12	
3	4	12	
4	5	12	
5	6	12	
Total		60	

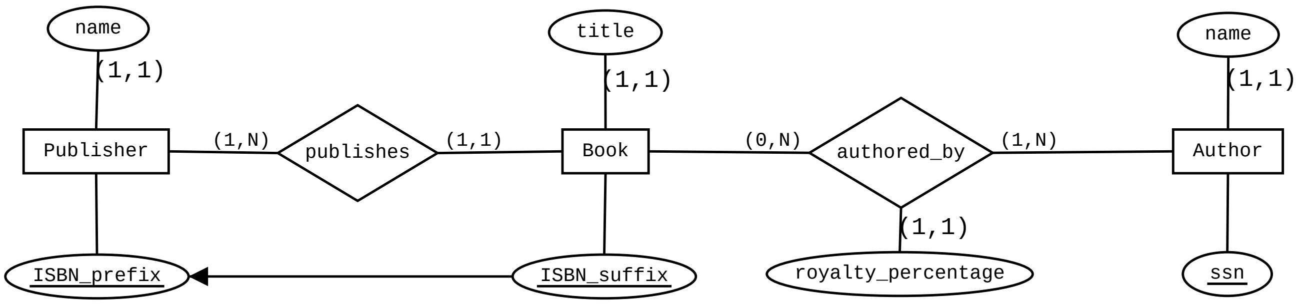
1. [**12 Points**] Given the attributes B, D, I, O, S, and Q with functional dependencies $S \rightarrow D$, $I \rightarrow B$, $I\,S \rightarrow Q$, and $B \rightarrow O$,

   (a) Show a lossless decomposition of the full set of attributes (B, D, I, O, S, and Q) into a collection of 3NF tables, preserving the dependencies. You needn't show the tables in SQL syntax, but at least indicate a name for each table for use in the following parts.

   (b) In each of your tables, which columns should be marked as the primary key?

   (c) For any columns that serve as a foreign key, give the following facts: what table the columns appear in as a foreign key, which columns they are, and which other table they reference.

2. [**12 Points**] Write a good set of SQL `create table` statements for the following E-R diagram. You needn't include `foreign key`/`references` information, but should designate the `primary key` of each table, and for any other column that must be `not null`, you should indicate that. Note that Book is a weak entity: `ISBN_suffix` only uniquely determines a book when used together with `ISBN_prefix`. You can make every column be of type `text`, for simplicity, except `royalty_percentage`, which should be `real`.



3. [**12 Points**] Recall that the `movie_info` table has columns for `id`, `title`, `director`, and `year_made`. Suppose we add two other tables:

```
create table stores (store_num int primary key,
                     location text not null);

create table holdings (store_num int not null references stores,
                       movie_id int not null references movie_info,
                       copies int not null check(copies > 0),
                       primary key (store_num, movie_id));
```

Show how to create a view, `mankato_movies`, which is just like `movie_info` except that it is restricted to just those movies of which the Mankato store holds copies.

4. [**12 Points**] Draw an E-R diagram corresponding to the following `create table` statements. Be sure to include cardinalities. *Hint:* first figure out which tables represent entities and which represent relationships. Similarly, for those tables that represent entities, figure out which columns represent identifiers, which represent non-identifier attributes, and which represent relationships with other entities.

```
create table A (a1 int primary key,
                a2 int not null unique references B);

create table B (b1 int primary key,
                b2 int not null);

create table C (c1 int references B,
                c2 int references D,
                primary key (c1, c2));

create table D (d1 int primary key,
                d2 int references A);
```

5. [**12 Points**] For each of the following scenarios, state whether it could happen:

   (a) under the "read committed" isolation level but not under the "serializable" isolation level,

   (b) under both "read committed" and "serializable," or

   (c) under neither "read committed" nor "serializable."

   (Nothing can happen only under serializable isolation level, since any isolation level is permitted to provide more isolation than it guarantees it will.)

   Each scenario involves two transactions, T1 and T2, manipulating two accounts, A and B. Before each scenario starts, accounts A and B each have balances of 100. You can assume that T2 in each case uses the serializable isolation mode—the question is all about T1. The scenarios are as follows:

   (a) T1 reads A's balance and finds 100. Transaction T2 then transfers 20 from A to B and commits. T1 then reads B's balance and finds 120. Finally, T1 commits.

   (b) T1 reads A's balance and finds 100. Transactions T2 then transfers 20 from A to B and commits. T1 then reads B's balance and finds 100. Finally, T1 commits.

   (c) T1 reads A's balance and finds 100. Transactions T2 then transfers 20 from A to B. T1 then reads B's balance and finds 120. T2 then decides to rollback. Finally, T1 commits.

   (d) T1 reads A's balance and finds 100. Transactions T2 then transfers 20 from A to B. T1 then reads B's balance and finds 120. T2 then commits. Finally, T1 commits.