MCS-274 Intra-term Exam 1

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. 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 50 minutes to work.

Write the answer to each problem on the page on which that problem appears. You may also attach additional paper, which should be labeled with your test number and the problem number.

You must sign the honor pledge below and abide by it.

Printed name: _____

On my honor, I pledge that I have not given, received, nor tolerated others' use of unauthorized aid in completing this work.

Signature for above honor pledge: _____

Problem	Page	Possible	Score
1	2	20	
2	3	20	
3	4	20	
4	5	20	
5	6	20	
Total		100	

1. [**20 Points**] A copy of the `create table` statements for the ships database is provided with this test. Write SQL queries that would answer each of the following questions using that database. Do not include any table in the `from` clause of a query unless the query makes use of the table. You may assume that there are no `null` values in any of the tables.

   (a) For each ship for which all the relevant data is available, list the name of the ship, its country, and the battle at which it was sunk.

   (b) For each ship for which we have no record of the ship being sunk, list the name of the ship and the ship's country. Your answer should use the NOT IN operation. You may limit your attention to ships for which we know the country.

   (c) Again list all those combinations of a ship's name and country such that we have no record of the ship being sunk. This time, your answer should use the NOT EXISTS operations. You may again limit your attention to ships for which we know the country.

2. [**20 Points**] A copy of the `create table` statements for the ships database is provided with this test. Write SQL queries that would answer each of the following questions using that database. Do not include any table in the `from` clause of a query unless the query makes use of the table. You may assume that there are no `null` values in any of the tables.

   (a) List the classes that contain ships and, for each one, how many ships they contain.

   (b) List the classes that contain four or more ships and, for each one, how many ships they contain.

   (c) List the classes that contain ships and, for each one, how many ships they contain. The list should be in order of decreasing number of ships. Within any given number of ships, the list should be in alphabetical order by class name.

3. [**20 Points**] A copy of the `create table` statements for the ships database is provided with this test. Write SQL queries that answer the following question using that database: What are the names of the ships that were launched in the earliest year included in the ships table?

   Do not include any table in the `from` clause of a query unless the query makes use of the table. You may assume that there are no `null` values in any of the tables.

   (a) Give an answer that makes use of MIN.

   (b) Give a different answer that makes use of ALL.

   (c) Give a different answer that makes use of ANY.

4. [**20 Points**] Suppose R and S are as shown:

R =

A	**B**	**C**
a1	b1	c1
a2	b2	c1
a1	b1	c2

S =

B	**C**	**D**
b2	c1	d1
b1	c2	d2
b1	c2	d3
b1	c3	d3

(a) Show a similar table for the relation R ⋈ S.

(b) Write an SQL query that does not contain a JOIN operator but nonetheless as closely as possible replicates the meaning of the relational algebra expression.

(c) Returning to the first part of this problem, what row(s), if any, would you add to your table if the ⋈ were changed to a natural full outer join?

5. [**20 Points**]

(a) Draw a table, S, so that R $\bowtie$ S is the same as the selection $\sigma_{\mathrm{n}>1 \text{ AND } \mathrm{n}<4}(\mathrm{R})$. All you know about the table R is that it has a column, n, containing integers. (It may have other columns too.)

(b) Write a relational algebra expression, not containing the $\bowtie$ symbol, that is equivalent to $\mathrm{T} \bowtie_{\mathrm{T.a=U.b}} \mathrm{U}$.

(c) Suppose V is a relation such that $\pi_{\mathrm{a,b}}(\mathrm{V}) = \mathrm{V} \bowtie \mathrm{V}$. What can you conclude about V? For example, do you know anything about how many rows (tuples) or columns (attributes) it has? Can you say anything about the contents of the rows or the names of the columns?