Final Examination
Course 1-02-324: Database Systems
Exam A

Kinneret College School of Engineering

January 30, 2011 9:00am - 12:00pm

- Answer the following questions in English or Hebrew.
- You may not use any material or learning aids to the test.
- The number of points for each question is listed next to each one to indicate its weight.
- There are a total of **100** points on the test. You must answer all of the questions.
- Write all of your answers in the test booklet which you received.
- Marks made on the test sheets will not be counted or graded.
- You must return the test questions sheet at the end of the exam.
- You must use proper SQL or MS SQL Server syntax for all answers.

המכללה האקדמית כנרת
בעמק הירדן (ע"ר) בית הספר להנדסה

1 Entity Relational Diagrams (30 points)

Read the following story:

> A pizza store manages its customers and orders in a database. Each customer has a customer ID, name, and address. Customers can place orders for pizzas. Each order is placed by one customer and has an order ID and a delivery date/time. An order is for one or more pizzas. Each pizza has a pizza code (from a list of available pizza types - each with a code and a name), a size, and a price.
>
> The price is determined by many factors (bulk discounts, periodic sales, etc.), so the salesman enters in the price specially for each order (*e.g.* in order 34 a 50cm deep dish pizza cost 40 shekels and in order 35 it cost 42 shekels), but all prices must be non-negative. All pizzas of a single type and size will have the same price in a given order (*i.e.* all 50cm deep dish pizzas in order 21 will cost 40 shekels each).
>
> A single order can contain several size/type combinations (*e.g.* three 50cm deep dish, two 40cm deep dish, and four 40cm Sicilian style).

Based on the above story, perform the following actions:

(a) (15 points) Prepare an Entity-Relationship Diagram (ERD) based on the above story. Indicate all attributes, keys, constraints, aggregation, inheritance, or weak entities used (you should not need to use all of them). If you were unable to model some aspect of the story, explain why.

(b) (15 points) Prepare a set of SQL CREATE TABLE definitions based on the ERD you prepared in the previous part. Use correct SQL syntax. Include all primary keys, foreign keys, and constraints. In case you cannot represent a given constraint from the ERD, explain why.

2 Relational Algebra and SQL

Consider the following relational schema:

Sailors (<u>sid:`INT`</u>, sname:`VARCHAR(30)`, rating:`INT`, age:`INT`)
Boats (<u>bid:`INT`</u>, bname:`VARCHAR(30)`, color:`VARCHAR(10)`)
Reserves (<u>bid:`INT`, sid:`INT`, day:`DATE`</u>)

2.1 Relational Algebra and SQL (30 points / 10 points each)

Write relational algebra statements and SQL for each of the following queries:

(a) Show the names of all sailors who reserved at least two boats with different colors.

(b) Show the id and name of each boat which was reserved by each sailor at least once.

(c) Show the name and rating of each sailor who never reserved boat number 11.

2.2 Aggregate SQL Queries (15 points)

Write SQL for the following query:

(a) Show the boat color which was reserved the most (*e.g.* red, blue, etc.).

3 Triggers (15 points)

Consider the following modified sailors/reserves/boats schema:

Sailors (sid:`INT`, sname:`VARCHAR(30)`, rating:`INT`, age:`INT`)
Boats (bid:`INT`, bname:`VARCHAR(30)`, color:`VARCHAR(10)`)
Reserves (bid:`INT`, sid:`INT`, day:`DATE`)
Canceled (sid:`INT`, bid:`INT`, day:`DATE`, age:`INT`, color:`VARCHAR(10)`)

The relation "Canceled" contains information about all canceled reservations including: the sailor ID and age of the sailor who canceled, the boat ID and color of the boat, and the day of the canceled reservation. For example, if Horatio (sid = 11, age = 25) canceled a reservation for the boat Interlake (bid = 21, color = red) on January 26, 2011, the relation Canceled will store a new row:

sid	bid	day	age	color
11	21	26/1/2011	25	'red'

(a) (15 points) Write an MS SQL Server trigger which automatically updates "Canceled" whenever a reservation is **deleted** from Reserves. Make sure your trigger fires only when a row is deleted from Reserves (not on updates) and doesn't attempt to insert lines which would violate the primary key constraint of Canceled. If the deletion would cause such an illegal insert, the trigger should just complete successfully without performing any action (and without aborting the action). Assume only one row is deleted from Reserves at a time.

(b) (Extra Credit + 5 points) Make your MS SQL Server trigger work even when multiple rows are deleted from Reserves.

4 Transactions (10 points)

Consider the following transaction schedule:

T1	T2	T3
R(A)		
	R(A)	
	W(B)	
R(B)		
		R(B)
		W(D)
	W(D)	
R(D)		
		W(D)
Commit		
	Commit	
		Commit

Is the above schedule *view serializable*? If yes, to what serial schedule is it *view equivalent*? If not, list all of the conflicts which prevent it from being view serializable.

5 Appendix: SQL and MS SQL Server Formats

5.1 Basic SQL Queries:

```
SELECT [select-list]
FROM [table list]
WHERE [condition list]
GROUP BY [grouping-list]
HAVING [group-condition]
```

5.2 Basic SQL Data Manipulation:

Insert `INSERT INTO Table1(field1, field2) VALUES (val1, val2)`

Delete `DELETE FROM Table1 WHERE [condition]`

Update `UPDATE Table1 SET [field assignments] WHERE [condition-list]`

5.2.1 Another version for insert:

```
INSERT INTO Table1 (field1, field2)
SELECT f1, f2 FROM Table2
```

5.3 SQL Data Definition:

Table Creation `CREATE TABLE [table name] ([field] [type], ..., PRIMARY KEY ([field list]), FOREIGN KEY ([field list]) REFERENCES [table])`

View Creation `CREATE VIEW [view name] ([field list]) AS [body] [WITH CHECK OPTION]`

5.4 MS SQL Server Stored Procedures and Triggers:

Trigger format:

```
CREATE TRIGGER triggerName ON
   tableName | viewName
   FOR | AFTER | INSTEAD OF
   [INSERT,] [UPDATE,] [DELETE]
   AS [IF UPDATE (columnName)] [batch-code]
```

Stored Procedure format:

```
CREATE PROCEDURE procedureName [(@parameter1
datatype1 [=DEFAULT], [@parameter2 datatype2
[= DEFAULT], ...])]
   AS batch-code
```

5.5 Relational Algebra

The general mapping from relational algebra to SQL:

Algebra	SQL Term
$\pi_{x,y}$	SELECT x, y
$R \times S$	FROM R, S
$\sigma_{x>y}$	WHERE $x > y$
$R \bowtie_c S$	FROM R, S WHERE c
$R \bowtie S$	S NATURAL JOIN R
$\rho(N(p1 \rightarrow o_1, p2 \rightarrow o_2), E)$	SELECT p_1 AS o_1, p_2 AS o_2 FROM E AS New WHERE …