The University Of New South Wales

Final Exam

June 2003

COMP9311

Introduction to Database Systems

Time allowed: **3 hours**
Total number of questions: **7**
Total number of marks: **90**
Textbooks, study notes, calculators, mobile phones, etc.
are **not** permitted in this exam.
Questions are **not** worth equal marks.
Answer **all** questions.
You can answer the questions in any order.
Start each Long Answer question on a new page in a script book.
If you use more than one script book,
fill in your details on the front of *each* book.
You may **not** take this question paper out of the exam.

Name:

Student#:

Multiple Choice Questions

Question 1

(20 marks total) Each of the multiple-choice questions in this part of the exam has *four* alternatives, only one of which is correct. Once you have chosen the correct alternative, circle your choice **in this question book**. In this part of the exam, each question is worth **2 marks**. There is a **penalty** of **-1 mark** for answering a question incorrectly. There is no penalty for not answering a question.

a) Consider the following ODL definition for classes describing `Department`s and the `Employee`s who work in them:

```
class Department {
   attribute string name;
   ...
   relationship Set<Employee> employees inverse Employee::worksIn;
};
class Employee {
   attribute string name;
   ...
};
```

Each employee works in just one department, and we wish to represent this 1:n relationship between departments and the employees that work in them. Which one of the following `relationship` definitions would complete the `Employee` class definition and capture this relationship?

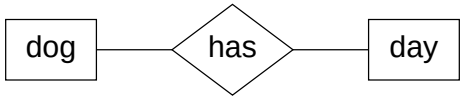
i) `relationship Department worksIn inverse Department::employees;`

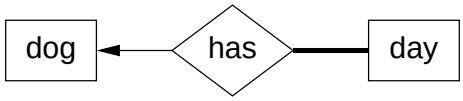
ii) `relationship Department worksIn inverse Set<Department>::employees;`

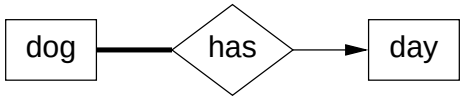
iii) `relationship Set<Department> worksIn inverse Department::employees;`

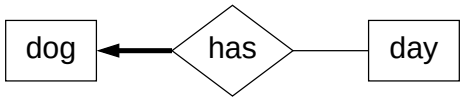
iv) `relationship Set<Department> worksIn inverse Set<Department>::employees;`

b) Which of the following ER diagrams most accurately captures the meaning of the english idiom "Every dog has its day"?

i)


ii)


iii)


iv)


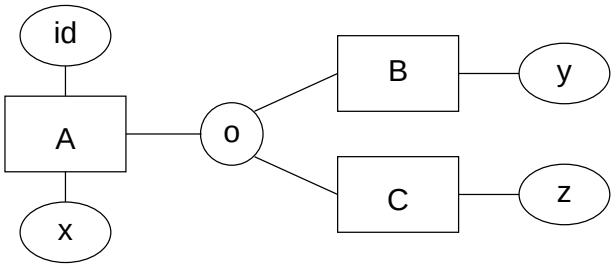
c) Consider the SQL query:

```
select X,Y,Z from R where Y = 2 and Z > 0
```

Which of the following relational algebra expressions gives a correct translation of this query?

i) $\sigma_{(X,Y,Z)}(\pi_{(Y=2 \wedge Z>0)} R)$
ii) $(X, Y, Z) \bowtie R_{(Y=2 \wedge Z>0)}$
iii) $\pi_{(X,Y,Z)}(\sigma_{(Z>0)}(\sigma_{(Y=2)} R))$
iv) $\pi_{(X,Y,Z)}(R \bowtie (Y = 2 \wedge Z > 0))$

d) Consider the following E/R diagram, showing subclasses `B` and `C` of superclass `A`:



Which one of the following is *not* a valid translation of this design into the relational model?

i) `A(id, isB, isC, x, y, z)`
ii) `A(id, x)  B(x, y)  C(x, z)`
iii) `A(id, x)  B(id, y)  C(id, z)`
iv) `A(id, x)  AB(id, x, y)  AC(id, x, z)  ABC(id, x, y, z)`

e) Consider the relation $R = ABCDE$ and the set F of functional dependencies on its attributes:

$$F = \{BC \to D,\ C \to A,\ BC \to A,\ A \to E\}$$

Which one of the following is the complete set of candidate keys of the relation R?

i) $\{BC\}$
ii) $\{A, D\}$
iii) $\{ABCDE\}$
iv) $\{BCD, AC, ABC, AE\}$

f) The `spool` meta-command in Oracle's's `SQL*Plus` shell ...

i) collects statistics about query cost
ii) prints the output of the previous query
iii) allows you to send query output to a file
iv) activates Oracle's's object-oriented extensions

g) Consider the relation $R = ABCD$ and the set F functional dependencies on its attributes:

$$F = \{AB \to C,\ AB \to D, C \to D\}$$

What is the highest normal form of R?

i) 1NF
ii) 2NF
iii) 3NF
iv) BCNF

h) A *sparse index* on attribute `A` ...

i) has a single entry for each data page and requires the data file to be sorted on A
ii) has multiple entries for each data page and requires the data file to be sorted on A
iii) has a single entry for each data page and does not require the data file to be sorted on A
iv) has multiple entries for each data page and does not require the data file to be sorted on A

i) Which of the following is *not* one of the so-called ACID properties required for a database system to handle transactions while preserving database integrity?

i) **A** = atomicity, either all transaction operations complete, or none do
ii) **C** = consistency, execution of transaction in isolation preserves database validity
iii) **I** = isolation, each transaction executes as if it the sole user of the DBMS
iv) **D** = duration, each transaction is completed in the minimal required time

j) Which of the following transaction schedules is conflict-serializable?

i) `T1:R(X) T2:R(X) T1:W(X) T2:W(X)`
ii) `T1:R(X) T2:W(X) T1:W(X) T2:R(Y) T3:W(X)`
iii) `T1:R(X) T1:R(Y) T1:W(X) T2:R(Y) T3:W(Y) T1:W(X)`
iv) None of the above schedules is serializable.

Long Answer Questions

The remaining questions make use of the following relational schema that implements an Online Bookstore application:

```
create table Books (
      isbn           char(10) primary key,
      title          text not null,
      authors        text not null,
      publisher      text not null,
      yearPublished  integer not null,
      price          real,
      qtyInStock     integer check (qtyInStock >= 0)
);

create table Customers (
      id             integer primary key,
      name           text not null,
      address        text not null,
      phone          text
);

create table OnLineCustomers (
      id             integer references Customers(id),
      username       char(10) not null,
      password       char(10) not null,
      email          text not null
);

create table Orders (
      id             integer primary key,
      customer       integer references Customers(id),
      datePlaced     date not null,
      discount       real not null
                       check (discount between 0 and 0.3),
      dateShipped    date,
      datePaid       date
);

create table OrderItems (
      orderid        integer references Orders(id),
      itemSeq        integer,
      bookisbn       char(10) references Books(isbn),
      quantity       integer,
      unitprice      real
);
```

Online Bookstore (cont)

The following requirements were considered in the design of the schema:

- for books, we must maintain their ISBN, title, author(s) and year of publication
- we also need to know how many copies of each book is currently in stock (i.e. in the warehouse and available for delivery) and what the current price is
- for customers, we need to know their name and address (for shipping), and (optionally) their phone number for urgent contact
- customers can deal with the bookstore either via a Web interface, or simply phone their orders through
- some customers deal with the bookstore exclusively by phone; for such customers the above information is all that is required
- online customers need to have a username and password, and must supply a contact email address, in addition the above customer information
- for orders, we need to remember who was the customer who place the order, when they placed it, when the order was shipped, whether they have paid
- orders are only shipped when all ordered items are available
- payments for orders must be received in full (no partial payments)
- customers receive a 5% discount on their order for every order over $100 made in the last 90 days (excluding the current order) up to a limit of 30%
- users pay the price that applied at the time they made their order
- customers may order multiple copies of a given book
- the company is trusting and may ship orders before payment is received

Question 2

(10 marks total)

Given the tables in the above schema, produce the ER diagram that they were most likely derived from. You do not need to show all attributes in your ER design, but you must show primary keys, and any relations which have attributes must have those attributes shown. You must also show all participation constraints and relationship cardinalities (in whichever notation you wish). If you need to make additional assumptions about the application in order to develop your ER diagram, then also include these.

Question 3

(10 marks total)

State the functional dependencies that hold in the Online Bookstore schema and prove (or disprove) that the schema is in third normal form (3NF). If you need to make any additional assumptions about the application in order to produce functional dependencies, please state them as well.

Question 4

(15 marks total)
Write SQL statements to answer each of the following queries:

a) How many different books does the bookstore have in stock?
(base the definition of "different book" on the title/author combination, not the ISBN)

b) Which books have multiple editions in stock?
(i.e. same title and author(s), but different publication year)

c) What are the email addresses of all customers with unpaid but shipped orders?

d) How many orders have been place by each on-line customer?
(you must include even on-line customers who have never placed an order)

e) Apply an "across-the-board" price increase of 10% to all books whose value is less than $100.

Question 5

(15 marks total)

a) Write a PL/SQL function `shippable(integer)` that takes an order identifier and returns a boolean result indicating whether all of the stock required to meet this order is available in the warehouse.

b) Write a PL/SQL function `orderAmount(integer)` that takes an order identifier and returns the total amount of money owed on that order, taking into account any discount.

c) Implement a trigger and a PL/SQL trigger procedure that ensures that the appropriate discount is included into an order when it is created.

Question 6

(10 marks total)
For each of the following queries, give one or more Oracle `CREATE INDEX` statements that would help the query to be processed more efficiently:

a)
```
select * from Orders where id = 7654321;
```

b)
```
select * from Orders where customer = 123456;
```

c)
```
select authors from Books where title like '%Database%';
```

d)
```
select title from Books where price between 50.00 and 100.00;
```

e)
```
select name,datePlaced from Orders,Customers
where  Orders.customer = Customer.id;
```

If you think that no index can help, or that one exists already that will help, state this instead of providing a `CREATE INDEX` statement.

Question 7

(10 marks total)
Consider five internal users of the above database: A, B, C, D, E. The following sequence of SQL `grant` statements is employed to set up access to the `Orders` table:

```
DBA: grant all on Orders to A,B with grant option;
A:   grant select,insert on Orders to C,E;
B:   grant update,delete on Orders to C with grant option;
C:   grant select on Orders to D;
C:   grant update on Orders to D;
B:   grant insert,delete on Orders to E;
```

Assume that the only privileges on `Orders` are `select`, `insert`, `delete`, `update`.

a) Did any of the `grant` statements fail? If so, state which one and explain why.

b) Draw a table showing which user has which privileges on the `Orders` table. Use the following format:

	select	insert	delete	update
A				
B				
C				
D				
E				

Place a tick in a square to indicate that a user has the privilege and a cross in a square to indicate that they do not.

c) Draw the table again to show the new set of privileges after execution of the statements:

```
A: revoke select from C restrict;
B: revoke update from C cascade;
```