Université d'Ottawa
Faculté de genie

École d'ingénierie et de
Technologie de l'information

uOttawa
L'Université canadienne
Canada's university

University of Ottawa
Faculty of Engineering

School of Information
Technology and Engineering

CSI2132
Database I

Midterm Winter 2011: Marking Guidelines

Length of Examination: 80 minutes **February 15, 2011, 13h00**

Professor: Dr. Herna L Viktor

Family Name: _____

First Name: _____

Student Number: _____

Signature _____

This is a Closed Book examination. Answer all the questions, **in INK**.
You are allowed to bring a single sheet of notes (letter size), printed or written on both sides.
If you do not understand a question, clearly state an assumption and proceed.
No calculators or other electronic devices are allowed.
At the end of the exam, when time is up:

- Stop working and turn your exam upside down.
- Remain silent.
- Do not move or speak until all exams have been picked up, and a TA or the Professor gives the go-ahead to leave.

Answer all questions on the question sheet in the areas provided.
You can use the back of the question sheet pages for calculations and other work. Do not detach any pages. ***Good luck***!

For use by grader:

Part A (17)	Part B (23)	Total (40)

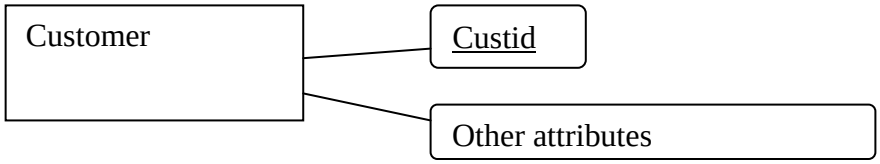
PART A: Conceptual Modeling [17 marks]

Note: Other correct examples were also acceptable.

Consider the *Swifty* world-wide package delivery company that specializes in delivering diamonds and other precious stones. You are asked to create a database for this enterprise. The database must be able to keep track of customers (who ship items) and customers (who receive items); some customers may do both. Each package must be identifiable and, importantly, trackable in real time. That is, the database should store the current location of a package together with its history of locations. Locations include trucks, planes, motorcycles, airports and warehouses. Each location has an employee who is responsible for the package, the so-called package handler. Typically, the handler is accompanied by a security guard. All employees are full-time, and have benefits including mandatory health and dental care and, in the case of the security guards, mandatory life insurance.

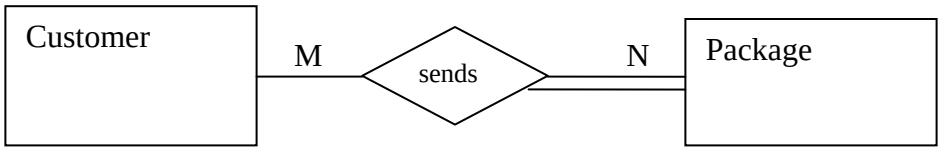
1. First, explain what a key constraint ensures and then draw an example that occurs in the EER diagram of the Swifty database. (1+1)

Textbook p.68, 208. A key constraint ensures that all tuples are uniquely identifiable.



2. First, explain what a total participation constraint ensures and then draw an example that occurs in the EER diagram of the Swifty database. (2+2)

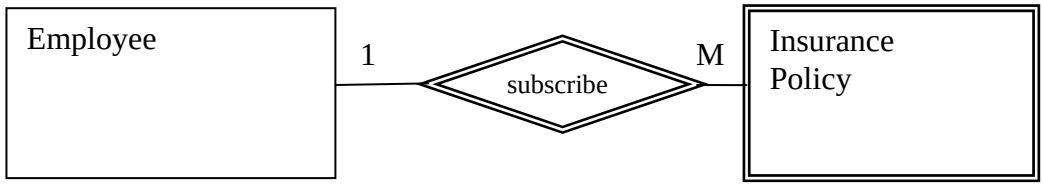
Textbook p.217. Every entity in the total set E1 must be related to another in the entity E2.



A Package (E1) must be associated with a Sender (E2).

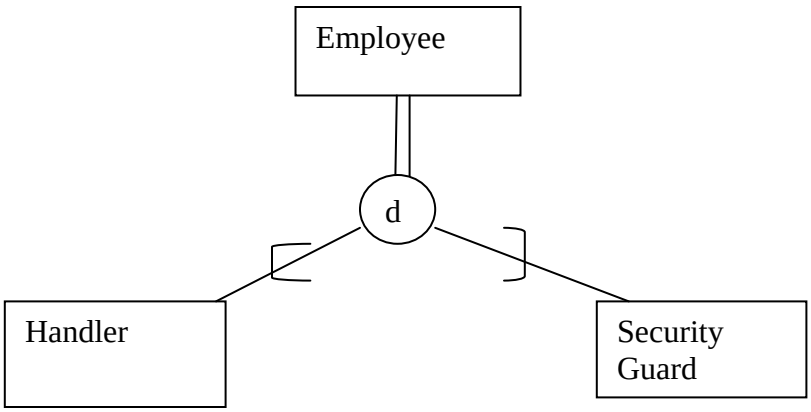
3. First, explain what a weak entity is and then draw an example that occurs in the EER diagram of the Swifty database. (2+2)

Textbook p.219. A weak entity W is existence dependent on an owner entity O. That is, W would not exist if O did not exist. Also, the weak entity has a partial key, which is combined with the primary key of O.



4. Draw a specialization hierarchy that exists in the EER diagram of the Swifty database. (4)

Textbook p.254



5. There are four different ways to map a specialization in EER models to the relational model. Which one of these options would you choose for the specialization you have shown above? Motivate your answer. (1+2)

p. 294: four options…

The second option, namely multiple (two) relations, subclasses only, would work best here.
This is because these subclasses are total and disjoint.

So, we would create two tables for Handlers and SecurityGuards.

Part B: The Relational Model and SQL [23 marks]

Answer the following questions regarding collectors that collect postage stamps for a hobby. You are provided with the following database schema, where the primary keys are underlined.

`Collector(CollectorID, lastname, firstname, date-of-birth, email, address, Country, postal-code, phone)`

`Order(CollectorID, StampID, date-of-order, date-shipped, quantity, total-charge)`

`Stamp(StampID, name, description, size, price, CatalogID)`

`Catalog(CatalogID, Country, date-of-issue)`

1. Explain, by using an example from the postage stamp database schema, **the difference between** a primary key, candidate key and super key. (3)

Textbook p.68
Superkey: Any combination of attributes that uniquely identify a tuple (e.g. CollectorID, lastname)

Candidate key: A minimal superkey (e.g. email)

Primary key: A candidate key that we choose (e.g. CollectorID)

2. In the current design, a collector would not be able to re-order a stamp. Explain how you would correct this mistake in the design. (3)

This is due to the primary key of the Order table.

A solution is to introduce a new primary key OrderID, and then make CollectorID and StampID foreign keys.

```
CREATE TABLE Stamp (
   StampID    integer,
   Name string,   Description  string,  size string,  price currency,
   CatalogID  integer,
   PRIMARY KEY (StampID),
   FOREIGN KEY (CatalogID) REFERENCE Catalog
         ON DELETE SET TO NULL
         ON UPDATE CASCADE);
```

3. Consider the above code, which is used to create the *Stamp* relation. Explain how you would change the `CREATE TABLE` statement to ensure that the *price* attribute is never left empty. (2)

Change to: price currency NOT NULL

4. Consider, again, the code to create the *Stamp* relation. Explain *when* the following two phrases will be executed and *what* they aim to ensure. (4)
   a. `ON DELETE SET TO NULL`
   b. `ON UPDATE CASCADE`

a.	If a Catalog is removed, then the CatalogID in the Stamp relation is set to NULL (empty) So, this implies that we do not associate this Stamp to any Catalog. Obviously, the Stamp needs to be associated with a new Catalog. In this case `ON DELETE RESTRICT` or `ON DELETE NO ACTION` would actually have been better.
b.	If the CatalogID of a Catalog, in the Catalog table, is changed, then changes are automatically propagated to the Stamp table. This ensures that a Stamp will always only appear in an existing Catalog.

5. The following SQL code fragment aims to list the names and emails of the collectors who ordered Canadian stamps that are priced higher than $1,000. The names and emails of the collectors should be displayed by their country of residence. Please correct the fragment. (4)

```
SELECT C.email, C.lastname, C.firstname
FROM Collector C, Order O, Stamp S, Catalog G
WHERE S.price > 1000 AND G.Country = “Canada”;
```

```
SELECT C.email, C.lastname, C.firstname
FROM Collector C, Order O, Stamp S, Catalog G
WHERE C.collectorID = O. CollectorID AND O.StampID = S.StampID
  AND S.CatalogID = G.CatalogID
  AND S.price > 1000 AND G.Country = “Canada”
GROUP BY C.Country;
```

6. What does the following SQL code fragment compute? (2)

```
SELECT C.Country, C.date-of-issue
FROM Catalog C
WHERE  (SELECT count(*)
        FROM Stamp S
        WHERE S.CatalogID = C.CatalogID) >= 5;
```

It displays the countries and data of issues of all catalogs that contain more than 4 stamps.

7. Give the SQL to find the full names and emails of all collectors who bought Estonian stamps in 2010, but *never* bought any Icelandic stamps. (5)

```
SELECT C.lname, C.fname, C.email
FROM Collector C, Order O, Stamp S, Catalog G
WHERE C.collectorID = O. CollectorID AND O.StampID = S.StampID
  AND S.CatalogID = G.CatalogID
  AND G.country = 'Estonia' AND G.Date-of-issues LIKE '%2010'
  AND C.CollectorID NOT IN
     (SELECT C2.CollectorID
     FROM Collector C2, Order O2, Stamp S2, Catalog G2
     WHERE C2.collectorID = O2. CollectorID AND O2.StampID = S2.StampID
          AND S2.CatalogID = G2.CatalogID
          AND G2.country = 'Iceland');
```

Note: Other correct solutions are, of course, also acceptable.