Université Libre de Bruxelles
Cours "Systèmes d'information distribués"
Année Académique 2000-2001
Examen de Seconde Session

1. Deductive Databases

The deductive database of an airline company has the following base predicates.

- trip(TripNo, From, To, Price) representing the fact that trip TripNo connecting the destinations From and To costs a Price.
  An example is as follows
  trip('IB750', 'Bruxelles', 'BsAires', 2000).
- tripSegment(TripNo, SegmNo, From, To, NoMiles) representing the individual segments of a particular trip, each having a number of miles.
  For example, the previous trip could have the following segments
  tripSegment('IB750', 1, 'Bruxelles', 'Madrid', 1000).
  tripSegment('IB750', 2, 'Madrid', 'SaoPaolo', 3400).
  tripSegment('IB750', 3, 'SaoPaolo', 'BsAires', 1200).
- dailyTrip(TripNo, Date, PlaneNo, TotalSeats) representing the fact that trip TripNo will be done at a particular Date by plane PlaneNo having a total number of seats.
  An example is as follows
  trip('IB750', 1/1/2001, 99765, 350).
- dailyTripSegment(TripNo, SegmNo, Date, AvailableSeats) representing the fact that the segment of a trip realized at a particular date has a number of available seats.
  For example, the previous trip could have the following daily trip segments
  dailyTripSegment('IB750', 1, 1/1/2001, 312).
  dailyTripSegment('IB750', 2, 1/1/2001, 350).
  dailyTripSegment('IB750', 3, 1/1/2001, 285).

Define the following predicates using Datalog rules:

a) tripIsFull(T, D) representing the fact that trip T at date D has no available seats in **at least one** of its segments.
b) nbrAvailableSeats(T, D, N) representing the fact that trip T at date D has N available seats, where N is the **minimun** of the available seats in all its segments.
c) popularSummerDestination(From, To) representing the fact that **all** trips between 1/7/2001 and 31/8/2001 having as destinations From and To are full.
d) ordListDestinations(T, L) representing the fact that L is the ordered list of destinations of trip T. In the above example the answer will be
   ordListDestinations ('IB750', ['Bruxelles', 'Madrid', 'SaoPaolo', 'BsAires']).
e) numberMiles(T, N) representing the fact that N is the sum of miles of all segments of trip T.

2. Temporal Databases

Consider the following TSQL2 schema belonging to a real estate company.

Property(propNo, street, number, town, zip, country, description)
Client(clientNo, firstName, lastName, street, number, town, zip, country, telephone, fax)
Owns(clientNo, propNo, percentage) AS VALID STATE DAY
    clientNo references Client.clientNo
    propNo references Property.propNo
Rents(clientNo, propNo, monthlyAmount, usage) AS VALID STATE DAY
    clientNo references Client.clientNo
    propNo references Property.propNo

where :

- Table Property contains the properties that the company rents.
- Table Client contains the persons owning and renting properties, since these two roles are not exclusive.
- Table Owns relates owners and properties. Attribute percentage is less than 100% if the property is shared among several owners.
- Table Rents keeps the clients renting a property and the monthly amount to be payed. Attribute usage, having as values "commercial" or "residence", establishes whether the property is used for commercial activities or for living.

Write the following queries

a) Give the first name, last name, and address of clients that currently own a property that is currently rented and the location ends before 1/08/2001.
b) Give the property number and address of properties located at Brussels and such that all rents during the year 2000 have been for commercial usage.
c) Give the property number and address of properties that have been unoccupied at least 3 months between two rentals.
d) Give for each owner the client number and the time interval during which he owned exactly one property (independently of the percentage).
e) Give the property number and address of properties whose monthly amount of rent never decreased.

3. Active Databases

Consider the following tables in a relational database of an airline company.

AirplaneType (Type, Constructor, Power, NbPlaces)
Mecanic (MecName, Address, Telephone)
Airplane (RegistrationNo, PurchaseDate, Type, NbRepairs)
    Type references AirplaneType.Type
RepairAccreditation(Mecanic, Type, FromDate)
    Mecanic references Mecanic.MecName
    Type references AirplaneType.Type
Repair(Airplane, Date, Mecanic, Purpose, Duration)
    Airplane references Airplane.RegistrationNo
    Mecanic references Mecanic.MecName

Consider the following integrity constraints.

- A mecanic can realize a repair only if s/he has an accreditation to repair the type of the airplaine.
- Due to the number of mecanics available, there cannot be two airplanes of the same type repaired at the same date.

Consider also the following derived rule.

- Attribute NbRepairs of Airplane is a derived attribute computed as the total number of repairs of that airplane in table Repair.

a) For each one of the above constraints:
– determine all the events that may cause a violation of the constraint, and
– derive in Starburst a trigger that rollbacks the transaction when one of these events occurs.
b) Determine all the events that may cause a recomputation of the derived attribute. Derive in Starburst a trigger that computes the attribute when one of these events occurs.

4. Object Databases

Consider the following ODMG schema storing the program of the films showing in movie theaters.

```
Struct Address { string street; string number; string zip; string town; }
class Film (extent Films) {
        attribute String name;
        attribute String countryOfOrigin;
        attribute Date releaseYear;
        attribute Interval duration;
        relationship list<Star> actors inverse Star::filmsPlayed;
        relationship Star director inverse Star:: filmsDirected;
        relationship list<FilmShowing> showings inverse FilmShowing:: film;
}
class Star (extent Stars) {
        attribute String name;
        attribute Date dateOfBirth;
        attribute String placeOfBirth;
        relationship list<Film> filmsPlayed inverse Film::actors;
        relationship list < Film> filmsDirected inverse Film::director;
}
class MovieTheater (extent MovieTheaters) {
        attribute String name;
        attribute Address address;
        attribute String phone;
        relationship list<FilmShowing> showings inverse FilmShowing::theater;
}
class FilmShowing (extent FilmShowings) {
        attribute Date date;  attribute Time time;
        attribute Integer roomNo;
        attribute Integer totalSeats; attribute Integer availableSeats;
        relationship Film film inverse Film::showings;
        relationship MovieTheater theater inverse FilmShowing::showings;
}
```

Write in OQL the following queries.

a) Give the name and address of all movie theaters showing the film "Billy Elliot" today (27/02/2001) and for each theater, the hour of the showings of that film.

Example of output

```
-----------------------------------------------
Kinepolis (1020 Bruxelles-Brussel)
16.45 19.45 22.15
-----------------------------------------------
U.G.C. Toison d'or (1060 Bruxelles-Brussel)
16.45 19.15 21.40
-----------------------------------------------
Vendôme (1050 Bruxelles-Brussel)
15.20 17.30 19.40 21.50
-----------------------------------------------
```

b) Give for each film, the film name as well as the name and address of all theaters showing that film today (27/02/2001).

Example of output

```
-----------------------------------------------
BILLY ELLIOT
Kinepolis (1020 Bruxelles-Brussel)
U.G.C. De Brouckère (1000 Bruxelles-Brussel)
U.G.C. Toison d'or (1060 Bruxelles-Brussel)
Vendôme (1050 Bruxelles-Brussel)
-----------------------------------------------
DANCER IN THE DARK
Actor's Studio (1000 Bruxelles-Brussel)
-----------------------------------------------
```

c) Give the name of the directors who appear as actors in at least one film.

d) Give the name of French actors who have been directed by at least one American director.

e) Give for each film the total number of persons that have seen the film in January 2001.