CSE 344 Homework 4: Relational Algebra, Datalog, Relational Calculus

Objectives:
To understand how SQL queries are translated into the relational algebra. To master writing relational queries in a logic formalism, like datalog and relational calculus.

Reading assignments:
2.4, 5.1, 5.2, 5.3, 5.4

Assignment tools:
None needed

Due date:
Thursday, February 13, 2013, at 11:59 pm. Turn in your answers here.

What to turn in:
`hw4-answers.{txt|pdf|doc}`

Consider the following relational schema:

```
Employee(eid, name, office)
Manager(eid, mid)
```

Each employee has a unique key, eid. An employee may have several managers, who are, in turn, employee: both attributes eid and mid in `Manager(eid, mid)` are foreign keys to `Employee`.

For each of the queries below, write it in the relational calculus, in datalog, and in the relational algebra. You should return three answers: (a) a relational calculus expression **(6 points)** (b) a query in datalog+negation **(7 points)** (c) a relational algebra plan **(7 points)**

1. Write a query that retrieves all employees that have two or more managers. Your query should return the eid's and the names.
2. An independent employee is an employee without a manager. (For example, the CEO is independent.) Write a query that retrieves all independent employees; you should return their eid and their names.
3. Retrieve the office of all managers of the employee called 'Alice'. If there are multiple employees called Alice, or if one of them has several managers, you have to return all their offices.
4. Find all managers for which all the employees they manage share the same office. Your query should return their eid, their name, and the office where all their managed people are located.
5. A manager is an employee who manages at least one other employee. A second-level manager is a manager who manages only managers. Write a query to return all second-level managers; your query should return their eid's and their names.

Extra Credit (2 points):
Write a datalog query that finds all employees that report directly or indirectly to 'Alice'. That is, your query should return all employees managed by 'Alice', all employees managed by the employees managed by 'Alice', etc.