

# Section 5 – RC & Datalog

CSE 344

## Question 1

Consider the following database schema:

*Neighbors*(name1, name2, duration)

*Colleagues*(name1, name2, duration)

**Write datalog query that returns all neighbors who do not have any colleagues in common:**

$\text{NonAnswers}(n1, n2) :- \text{Neighbors}(n1, n2, -), \text{Colleagues}(n1, c, -),$

$\text{Colleagues}(n2, c, -)$

$\text{A}(n1, n2) :- \text{Neighbors}(n1, n2, -), \text{NOT NonAnswers}(n1, n2)$

## Question 2

Relational Calculus Queries –same schema as above–

**Write an RC query to find all people who have a neighbor that has a colleague:**

$\text{A}(x) = \exists y. \exists z. \exists n. \text{Neighbors}(x, y, n) \wedge \exists m. \text{Colleagues}(y, z, m)$

**Write an RC query to find all people who have only neighbors that are also their colleagues:**

$\text{A}(x) = \exists m. \exists n. \text{Neighbors}(x, m, n) \wedge (\forall y. \exists o. \text{Neighbors}(x, y, o) \Rightarrow \exists d. \text{Colleagues}(x, y, d))$

**Write a datalog query to find all people who have only neighbors that are also their colleagues:**

$\text{C}(x) :- \text{Neighbors}(x, y, -), \text{Colleagues}(-, -, d1), \text{not Colleagues}(x, y, d1)$

$\text{A}(x) :- \text{Neighbors}(x, y, -), \text{not C}(x)$

**Write an RC query to find all people who have only neighbors that have at least one colleague.**

$\text{A}(x) = \exists m. \exists n. \text{Neighbors}(x, m, n) \wedge \forall y. (\exists o. \text{Neighbors}(x, y, o) \Rightarrow \exists z. \exists d. \text{Colleagues}(y, z, d))$

### Question 3

Consider the following database schema:

*Person(pid, name)*

*Trusts(pid1, pid2)*

Answer each question below by writing a query in non-recursive datalog with negation.  
Return the person id and the name.

**Write a datalog query to find the people who trust everyone except themselves:**

$S(p) :- \text{Person}(p, -), \text{Person}(q, -), \text{not Trusts}(p, q), p \neq q$   
 $S(p) :- \text{Person}(p, -), \text{Trusts}(p, p)$   
 $A(p, n) :- \text{Person}(p, n), \text{not } S(p)$

**A “loner” is a person who trusts no-one but himself. Write a datalog query that returns all loners:**

$NA(p) :- \text{Trusts}(p, x), p \neq x$   
 $A(p, n) :- \text{Person}(p, n), \text{Trusts}(p, p), \text{not } NA(p)$

**A “loyal” is a person who trusts only those who trust him. Write a datalog query that returns all loyal people.**

$NA(p) :- \text{Trusts}(p, x), \text{not Trusts}(x, p)$   
 $A(p, n) :- \text{Person}(p, n), \text{not } NA(p)$

**A “ruler” is a person who trusts only those who trust only him. Write a datalog query that returns all rulers.**

$NA(p) :- \text{Trusts}(p, x), \text{Trusts}(x, y), p \neq y$   
 $A(p, n) :- \text{Person}(p, n), \text{not } NA(p)$

**Write an SQL query that returns all rulers.**

```
SELECT p.pid, p.name
FROM Person p
WHERE not exists
  (SELECT *
   FROM Trusts t1, Trusts t2
   WHERE t1.pid1 = p.pid and t1.pid2 = t2.pid1 and t1.pid1 <> t2.pid2)
```