CSE344 19AU Worksheet
Lecture 7: Relational Algebra

The following problems use these table definitions:

```sql
CREATE TABLE Person (  
    pid  INT PRIMARY KEY,       -- person ID  
    name VARCHAR(100));       -- person name

CREATE TABLE Email (  
    eid     INT PRIMARY KEY,       -- email ID  
    pidFrom INT REFERENCES Person, -- email sender  
    length  INT);                  -- email char length

CREATE TABLE EmailTo (  
    eid   INT REFERENCES Email,   -- email ID  
    pidTo INT REFERENCES Person,  -- email recipient  
    PRIMARY KEY (eid, pidTo));
```

Relational algebra operators:
Union $\cup$    Difference $-$    Selection $\sigma$    Projection $\pi$    Join $\Join$
Rename $\rho$    Duplicate elimination $\delta$    Grouping and aggregation $\gamma$    Sorting $\tau$

A witnessing problem: List the pid of people who wrote the longest emails to themselves and the length of the emails.

```sql
SELECT E1.pidFrom, MAX(E2.length)  
FROM Email E1, EmailTo T1, Email E2, EmailTo T2  
WHERE E1.eid = T1.eid AND  
    T1.pidTo = E1.pidFrom AND  
    E2.eid = T2.eid AND  
    T2.pidTo = E2.pidFrom  
GROUP BY E1.pidFrom, E1.length  
HAVING E1.length = MAX(E2.length);
```

Draw the RA tree for the query
The following problems use these table definitions:

CREATE TABLE Person (
    pid INT PRIMARY KEY, -- person ID
    name VARCHAR(100)); -- person name

CREATE TABLE Email (
    eid INT PRIMARY KEY, -- email ID
    pidFrom INT REFERENCES Person, -- email sender
    length INT); -- email char length

CREATE TABLE EmailTo (
    eid INT REFERENCES Email, -- email ID
    pidTo INT REFERENCES Person, -- email recipient
    PRIMARY KEY (eid, pidTo));

Relational algebra operators:
Union ∪ Difference — Selection σ Projection π Join ⊗ Rename ρ Duplicate elimination δ Grouping and aggregation γ Sorting τ

A subquery problem: Find all emails where all of the recipients are named Alice.

SELECT E1.eid
FROM Email E1
WHERE NOT EXISTS (SELECT *
    FROM EmailTo E2, Person P
    WHERE E1.eid = E2.eid AND
    E2.pidTo = P.pid AND
    P.name != 'Alice');

Write the uncorrelated version of the query

SELECT E1.eid
FROM Email E1
WHERE E1.eid NOT IN (SELECT E2.eid
    FROM EmailTo E2, Person P
    WHERE E2.pidTo = P.pid AND
    P.name != 'Alice');

Draw the RA tree for the uncorrelated version of the query