Short Questions

Question 1. [5 marks]

Given two relations R1 and R2, where R1 contains N1 tuples, R2 contains N2 tuples, and N2 > N1 > 0, give the minimum and maximum possible sizes (in tuples) for the resulting relation produced by each of the following relational algebra expressions. In each case, state any assumptions about the schemas for R1 and R2 needed to make the expression meaningful:

(a) R1 ∪ R2. R1 and R2 need to be union compatible (same arity and same domain)

Min: N2, Max: N1+N2

(b) R1 ∩ R2 R1 and R2 need to be union compatible (same arity and same domain)

Min: 0, Max: N1

(c) R1 − R2 R1 and R2 need to be union compatible (same arity and same domain)

Min: 0, Max: N1

(d) R1 × R2

Min and Max is both N1 * N2

(e) R1 / R2

Assume attributes of R2 is subset of R1

Min and Max number is 0. Because number of tuples in R2 is greater than R1

Question 2. [5 marks]
Which one of the following constructs gives us the ability to write correlated subqueries in SQL? Explain your answer briefly.
a) EXISTS
b) IN
c) CASE expression
d) Subquery in FROM clause

Answer: EXISTS
Explanation should be provided

Schema for Relational Algebra and SQL queries:
Consider following schema for *Preyground* bus station, an intercity bus service in North America, serving thousands of destinations. *Preyground* has different types of buses which are driven by qualified drivers with valid licence for that type of bus.
Preyground has designed a database for bus routes information that also tracks the licence data for their bus drivers.
The database has following four relations with primary keys underlines and attribute domain listed besides the attribute name:

bus (bid: integer, bname: string, fuelrange: integer)
routes (rno: integer, departs: time, arrives: time, from: string, to: string, distance: integer, price: real)
employees (eid: integer, ename: string, salary: integer)
licensed (eid: integer, bid: integer)

- Relation *bus* has bid as primary key. It also has *bname* that has bus names (such as Shuttle bus, Minibus, Minicoach, Double-decker bus, Single-decker bus, Low-floor bus) listed.
- **Note:** *fuelrange* defines the maximum distance a bus can travel at a given speed without stopping for refueling.
- Relation routes has rno as primary key. The relation lists the bus routes, distance, with departure and arrival times and price for the routes.

- Relation `employee` is identified uniquely using `eid`. It has `ename` and `salary` as other attributes. The relation lists all kinds of employees including drivers, ticket agents, mechanics, baggage agents, bus cleaners.
- Relation `licensed` lists the employees licensed to drive a particular kind of bus given by `bid`. Every employee is licensed to drive some bus. Only drivers are allowed to operate buses.

Relational Algebra

Answer the following queries in Relational Algebra. If the query cannot be expressed in Relational Algebra, briefly explain the reason for it.

Question 3. [5 marks] Find the names of the drivers who are licensed for some Single-decker bus.

$$\pi_{ename}(\sigma_{bname\ =\ 'Single-decker\ bus'}(bus \bowtie licensed \bowtie employees))$$

Question 4. [5 marks] Find the names of the drivers who can drive buses with a range greater than 200 miles but are not licensed to drive any Double-decker bus.

$$\rho(R1, \pi_{eid}(\sigma_{fuelgrange} > 200(bus \bowtie Licensed)))$$

$$\pi_{ename}(Employees \bowtie (R1 - \pi_{eid}(\sigma_{bname} = 'Double-decker\ bus'(bus \bowtie licensed))))$$

Question 5. [5 marks] Find the id of the employees who are licensed to drive highest number of buses.

This cannot be expressed in relational algebra because there is no operator to count, and this query requires the ability to count up to a number that depends on the data.

Question 6. [5 marks] Find the id of the employees who are licensed to drive exactly three buses.

The approach behind this query is to first find the employees who are licensed for at least three buses (they appear at least three times in the licensed relation).

Then find the employees who are licensed for at least four buses. Subtract the second from the first and what is left is the employees who are licensed for exactly three buses.

$$\rho(R1, licensed)$$
$$\rho(R2, licensed)$$
$$\rho(R3, licensed)$$
$$\rho(R4, licensed)$$
$$\rho\Big(R5, \pi_{eid}\Big(\sigma_{(R1.eid = R2.eid = R3.eid) \wedge \begin{pmatrix} R1.bid <> R2.bid \wedge R1.bid <> R3.bid \wedge R2.bid \\ <> R3.bid \end{pmatrix}}(R1 \times R2 \times R3)\Big)\Big)$$

$$\rho(R6, \pi_{eid}(\sigma_{(R1.eid = R2.eid = R3.eid = R4.eid) \wedge}$$

$$(R1.bid <> R2.bid \wedge R1.bid <> R3.bid \wedge R1.bid <> R4.bid \wedge R2.bid <> R3.bid \wedge R2.bid < > R4.bid \wedge R3.bid <> R4.bid)$$

Question 7. [5 marks] Find the id of employees who drive every bus with fuelrange > 350.

$$(\pi_{eid,bid}\ licensed)/(\pi_{bid}\sigma_{fuelrange} > 350\ bus)$$

SQL

Answer the following queries in SQL.

Question 8. [5 marks] Find the bus ids of the buses that can be used on every *non-stop route* from Toronto to Cleveland.
Note: Non-stop route buses do not need to stop for fuel.

```
SELECT B.bid
FROM Bus B
B.fuelrange > ( SELECT MAX (R.distance)
                FROM routes R
                WHERE R.from = 'Toronto' AND R.to = 'Cleveland' )
```

Question 9. [5 marks] Find the names of drivers who are licensed to drive some Double-decker bus.

```
SELECT E.ename
FROM bus B, licensed L, employees E
WHERE B.bid = L.bid AND B.bname = 'Double-decker' AND E.eid = L.eid
```

Question 10. [5 marks] For each driver who is licensed for more than five kinds of buses, find the ids and maximum fuelrange of the buses for which he/she is licensed.

```
SELECT L.eid, MAX(B.fuelrange)
FROM Licensed L
JOIN Bus B ON L.bid = B.bid
GROUP BY L.eid
HAVING COUNT(*) > 5;
```

Question 11. [5 marks] Consider all the buses with fuelrange over 250 miles. Find the names of buses and average salary of all the drivers licensed to drives these buses.

```
SELECT AVG(E.salary) AS avgSalary
FROM employee E
WHERE E.eid IN (
    SELECT DISTINCT L.eid
    FROM licensed L
    JOIN bus B ON L.bid = B.bid
    WHERE B.fuelrange > 250
);
```

Question 12. [5 marks] Create a view named `SalaryDifference`. In this view, store the difference between the average salary of a bus driver and the average salary of all employees (including drivers). Using this view, find the name and salary of every nondriver whose salary is more than `SalaryDifference`.

```
CREATE VIEW SalaryDifference AS
SELECT
   (SELECT AVG(E.salary)
    FROM employees E
    WHERE E.eid IN (SELECT DISTINCT L.eid FROM licensed L))
   -
   (SELECT AVG(E.salary) FROM employees E)
AS diff;


SELECT E.ename, E.salary
FROM employees E
WHERE E.eid NOT IN (SELECT DISTINCT L.eid FROM licensed L)
  AND E.salary > (SELECT diff FROM SalaryDifference);
```