Question 1.  [6 marks]

Part (a)  [4 marks]

Suppose that

- $R_1$ is a relation with $t_1$ tuples and $a_1$ attributes.
- $R_2$ is a relation with $t_2$ tuples and $a_2$ attributes.
- $L$ is a list of $n$ attributes.
- $c$ is a boolean expression involving the attributes of $R_1$.

Assume that the expressions below are legal expressions of relational algebra. Fill in the table to indicate the size of the relation that is the result of each expression.

Solution:

<table>
<thead>
<tr>
<th>Expression</th>
<th>Number of tuples minimum</th>
<th>maximum</th>
<th>Number of attributes minimum</th>
<th>maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Pi_L R_1$</td>
<td>1</td>
<td>$t_1$</td>
<td>$n$</td>
<td>$n$</td>
</tr>
<tr>
<td>$\sigma_c R_1$</td>
<td>0</td>
<td>$t_1$</td>
<td>$a_1$</td>
<td>$a_1$</td>
</tr>
<tr>
<td>$R_1 \bowtie R_2$</td>
<td>$t_1 \times t_2$</td>
<td>$t_1 \times t_2$</td>
<td>$a_1 + a_2$</td>
<td>$a_1 + a_2$</td>
</tr>
<tr>
<td>$R_1 \bowland R_2$</td>
<td>0</td>
<td>$t_1 \times t_2$</td>
<td>$\max(a_1, a_2)$</td>
<td>$a_1 + a_2$</td>
</tr>
</tbody>
</table>

Part (b)  [2 marks]

Suppose $R$ and $S$ are relations. Which of the following statements are true? Circle one answer for each. Do not guess. There is 1 point for each correct answer, -1 for each incorrect answer, and 0 points if you leave the answer blank.

1. If $R$ and $S$ have no attributes in common, $R \times S = R \bowtie S$.
   True  False

2. If $R$ and $S$ have at least one attribute in common, it cannot be true that $R \times S = R \bowtie S$.
   True  False
Solution:

Part 1 is true.

Part 2 is also true, but only because the schemas of the two relations are necessarily different: $R \times S$ includes each common attribute twice, while $R \bowtie S$ does not. But it is possible to have an $R$ and an $S$ where the tuples that are included are the same. This occurs, for example, if on each common attribute, both relations have a single value and it’s the same value.

Question 2.  [8 marks]

Consider the following database:

\[
\begin{array}{ccc}
<table>
<thead>
<tr>
<th>P</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>
\end{array}
\]

\[
\begin{array}{cc}
<table>
<thead>
<tr>
<th>Q</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td></td>
</tr>
</tbody>
</table>
\end{array}
\]

Assuming set semantics, give the result (schema and data) returned by the following queries. Use the same tabular format as above; do not describe the result in English.

Part (a)  [2 marks]

\[(\Pi_C P - \Pi_C Q) \cap (\Pi_C P - \Pi_C (P \bowtie_5 Q))\]

Solution:

\[
\begin{array}{c}
<table>
<thead>
<tr>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
</tr>
<tr>
<td>3</td>
</tr>
</tbody>
</table>
\end{array}
\]

Part (b)  [2 marks]

\[T := \sigma_{P1.A < P2.A \land P1.C = P2.C}(\rho_{P1}(P) \times \rho_{P2}(P))\]

Answer := $\Pi_C P - \Pi_{P1.C} T$

Solution:

Strictly speaking, this query is ill-formed. The left operand of the set difference has attribute $C$, whereas the right operand has attribute $P1.C$. A rename would have fixed this.

\[
\begin{array}{c}
<table>
<thead>
<tr>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>2</td>
</tr>
</tbody>
</table>
\end{array}
\]
Part (c) [2 marks]
\[ T := (\Pi_A P \times \Pi_C Q) - (\Pi_{A,C} (P \Join Q)) \]
Answer := \Pi_A P - \Pi_A T

Solution:

Part (d) [2 marks]
\[ P_1(A, B, C) := P \]
\[ P_2(A, B, C) := P \]
\[ T := \sigma_{P_1.C = P_2.C \land P_1.B > P_2.B} (P_1 \times P_2) \]
Answer := \Pi_{A,P_1.B,P_1.C} T

Solution:

Again, strictly speaking, this query is ill-formed. The left operand of the set difference has attribute \( A, B, C \), whereas the right operand has attribute \( P_1.A, P_1.B, P_1.C \). A rename would have fixed this.
**Question 3.** [10 marks]

Consider the following schema for a hair salon. Keys are underlined.

- Clients(CID, name, phone).
  
  *CID* is the ID of a client, *name* and *phone* are their name and phone number.

- Staff(SID, name).
  
  *SID* is the ID of a staff member and *name* is their name.

- Appointments(CID, date, time, service, SID)
  
  *CID* is the ID of the client whose appointment it is, *date* and *time* indicate when the appointment happens, *service* is the name of the service they have at this appointment, and *SID* is the ID of the staff member providing the service for this appointment. *CID* is a foreign key on Clients and *SID* is a foreign key on Staff. That is, the following inclusion dependencies hold:

  - Appointments[CID] ⊆ Clients[CID], and
  - Appointments[SID] ⊆ Staff[SID].

Which of the following queries correctly find the name of every client who has not had a haircut in 2010? Circle one answer for each. **Do not guess.** There are 2 points for each correct answer, -1 for each incorrect answer, and 0 points if you leave the answer blank.

1. \[ A := (\Pi_{CID} \text{Clients}) - (\Pi_{CID} (\sigma_{date.year=2010 \land service=“haircut”} \text{Appointments})) \]
   
   \[ Answer := \Pi_{name} (A \bowtie Clients) \]
   
   Correct  Incorrect

2. \[ A := (\Pi_{CID,name} \text{Clients}) - (\Pi_{CID,name} (\sigma_{date.year=2010 \land service=“haircut”} (\text{Clients} \bowtie \text{Appointments}))) \]
   
   \[ Answer := \Pi_{name} A \]
   
   Correct  Incorrect

3. \[ A := (\text{Clients} \bowtie \text{Appointments}) - (\sigma_{date.year=2010 \land service=“haircut”} (\text{Clients} \bowtie \text{Appointments})) \]
   
   \[ Answer := \Pi_{name} A \]
   
   Correct  Incorrect

   **Note:** This query would inappropriately omit clients who have never had an appointment.

4. \[ A := (\Pi_{CID} \text{Clients}) - (\Pi_{CID} (\sigma_{date.year=2010 \land service≠“haircut”} \text{Appointments})) \]
   
   \[ Answer := \Pi_{name} (A \bowtie \text{Clients}) \]
   
   Correct  Incorrect

5. \[ A := (\Pi_{CID} (\sigma_{date.year=2010 \land service≠“haircut”} \text{Appointments})) \]
   
   \[ Answer := \Pi_{name} (A \bowtie \text{Clients}) \]
   
   Correct  Incorrect
Question 4.  [8 marks]

This question assumes the same schema as for question 3.

Write the following queries using only the basic Relational Algebra operators $\Pi, \sigma, \bowtie, \times, \cap, \cup, -, \rho$. Assume the set semantics (not bag semantics) for Relational Algebra.

1. CID of all clients who have never had an appointment for both a haircut and another, different, service on the same date.

   Solution:

   \[
   
   \begin{align*}
   Pairs &:= \rho_{A1}(Appointments) \times \rho_{A2}(Appointments) \\
   Have(CID) &:= \Pi_{A1.CID}(\sigma_{A1.CID=A2.CID \land A1.service="haircut" \land A2.service\neq"haircut" \land A1.date=A2.date}(Pairs)) \\
   Answer &:= (\Pi_{CID}Clients) - Have
   \end{align*}
   \]

2. Name and phone number of the client who had staff member Guilano’s first appointment.

   Solution:

   \[
   \begin{align*}
   Guiliano &:= \Pi_{CID,date,time}\sigma_{name="Guiliano"}(Appointments \bowtie Staff) \\
   Pairs &:= \rho_{G1}(Giuliano) \times \rho_{G2}(Giuliano) \\
   Beaten(CID, date, time) &:= \Pi_{G1,CID,G1.date,G1.time}(\sigma_{G1.date>G2.date \lor (G1.date=G2.date \land G1.time>G2.time)}(Pairs)) \\
   Answer &:= \Pi_{name}(\text{Giuliano} - \text{Beaten}) \bowtie Clients
   \end{align*}
   \]