For problems 1–4, we use the following relational schema:

- `Product(maker, model, category)`
- `PC(model, speed, ram, hd, price)`
- `Laptop(model, speed, ram, hd, screen, price)`
- `Printer(model, color, type, price)`

where attribute names should be self-explanatory. The constraints are as follows:

- `model` is the key for all relations.
- The only possible values of `category` are “PC”, “laptop”, and “printer”.
- The only possible values of `type` are “laser” and “ink-jet”.
- Every PC, laptop or printer must be referenced in relation `Product`.

**Problem 1 (10 marks)** Write an assertion statement for the following constraint: *Products of different kinds cannot have the same model number. For example, a PC and a laptop cannot have the same model number, nor can a printer and a laptop, etc.*

**Problem 2 (10 marks)** Write an assertion statement for the following constraint: *For every PC, there is a laser printer, and a color ink-jet printer, by the same maker.*

**Problem 3 (15 marks)**. Write the following query in relational algebra (you may use \(\pi, \sigma, \times, \cup, \setminus\) and renaming of attributes. Renaming of relations, and the linear notation from the text are not allowed).

Find PCs that are faster and have more ram than all the laptops by the same manufacturer (`maker`).

Do not assume that the assertions of the first two problems are valid.

**Problem 4 (15 marks)**. Write a relational algebra query that finds manufacturers (makers) whose laptops ought to be avoided. That is, for every laptop they make, there is another one, by a different manufacturer, that has a faster processor, more ram, a larger screen, and costs less.

The rules stated in Problem 3 apply.

**Problem 5 (10 marks)** Is the following schedule conflict serializable? Explain why. If it is, give an equivalent serial schedule.

<table>
<thead>
<tr>
<th></th>
<th>(T_1)</th>
<th>(T_2)</th>
<th>(T_3)</th>
<th>(T_4)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>write(X)</td>
<td>read(X)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>write(Y)</td>
<td>read(Y)</td>
<td>read(X)</td>
<td>read(Y)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>write(Y)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>write(X)</td>
</tr>
</tbody>
</table>
**Problem 6 (20 marks)** Consider the following database schema. Its attributes are

\[ A, B, C, D, E, F, G \]

and FDs are

\[
\begin{align*}
AB & \rightarrow CF \\
BG & \rightarrow C \\
AEF & \rightarrow C \\
ABG & \rightarrow ED \\
CF & \rightarrow AE \\
A & \rightarrow CG \\
AD & \rightarrow FEG \\
AC & \rightarrow B
\end{align*}
\]

Each of the questions is worth 5 marks.

1. Find the candidate keys and the prime attributes of this schema.
2. Compute the minimum cover of this schema.
3. Produce a 3NF decomposition.

**Problem 7 (20 marks)** A set \( X \) of attributes is an antikey if it is not a key, but every proper superset \( Y \) of \( X \) is a key (recall that \( Y \) is a proper superset of \( X \) if \( X \subset Y \) and \( Y \neq X \)). Is the following true: an attribute is prime if and only if it does not belong to at least one antikey?

If it is true, provide a proof; if it is false, give a counterexample. The maximum length of your answer is 13 lines in at least 11pt font, if you type, or equivalent for handwritten assignments. Longer answers will automatically receive 0 marks.