Logic in Computer Science

TIME ALLOWED : One and a Half Hour

INSTRUCTIONS TO CANDIDATES

Answer ALL questions.
1. Transform the following two formulas into DNF (disjunctive normal form).

- \((r \land (p \lor \neg q))\);
- \(\neg(p \land \neg(q \lor \neg r))\).

In your answer, state which laws (de Morgan, distributive law, complement law, etc) are applied.

(14 marks)

2. Consider the formula \(P = (p \land \neg(q \lor \neg(q \land p)))\). Check whether \(P\) is satisfiable using (a) truth tables and (b) the tableau algorithm from the lecture notes. In your answer for (b), show how the rules \(\implies \land, \implies \lor, \implies \neg, \implies \neg \land\) and \(\implies \neg \lor\) from the lecture notes are applied.

(26 marks)

3. Let \(S\) be the signature consisting of the unary predicate symbol lecturer, the binary predicate symbol teaches, and the individual constants Peter and John. Translate the following sentences into first-order predicate logic sentences over \(S\):

- Peter is a lecturer.
- Peter teaches John.
- Peter teaches somebody.
- Somebody teaches John.
- Everybody who teaches somebody is a lecturer.
- John does not teach anybody.

(24 marks)

4. Let \(S = \{R, Q, c\}\) be a signature, where \(R\) is a binary predicate symbol, \(Q\) is a unary predicate symbol, and \(c\) is an individual constant. Let

\[ \mathcal{F} = (D^\mathcal{F}, R^\mathcal{F}, Q^\mathcal{F}, c^\mathcal{F}) \]

be the \(S\)-structure defined by

- \(D^\mathcal{F} = \{c, d, e, f\}\);
- \(R^\mathcal{F} = \{(c, d), (d, e), (e, f)\}\);
- \(Q^\mathcal{F} = \{d\}\);
- \(c^\mathcal{F} = c\).

(a) Does \(\mathcal{F} \models Q(c)\) hold? Explain your answer.
(b) Does \(\mathcal{F} \models \exists x. Q(x)\) hold? Explain your answer.
(c) Does \(\mathcal{F} \models \forall x. \exists y. R(x, y)\) hold? Explain your answer.
(d) Does \(\mathcal{F} \models \forall x (Q(x) \to \exists y. R(x, y))\) hold? Explain your answer.
(e) Does there exist a variable assignment $a$ such that $(\mathcal{F}, a) \models R(c, x)$? Explain your answer.

(25 marks)

5. The following two notions are introduced in the lecture notes:

- $G$ is true in $\mathcal{F}$, in symbols $\mathcal{F} \models G$, where $\mathcal{F}$ is a structure and $G$ a first-order predicate logic sentence;
- $G$ follows from $X$, in symbols $X \models G$, where $X$ is a set of first-order predicate logic sentences and $G$ a first-order predicate logic sentence.

Define the second notion using the first one. Compare both notions regarding their complexity/decidability and their applications in computer science.

(11 marks)
Solutions

1. The transformations are as follows:
   - By complement law, \((r \land (p \lor \neg q))\) is equivalent to \((r \land (p \lor q))\) which is equivalent, by distributive law, to \(((r \land p) \lor (r \land q))\) which is in DNF.
   - By de Morgan law, \(\neg (p \land \neg (q \lor \neg r))\) is equivalent to \((\neg p \lor \neg (q \lor \neg r))\) which, by complement law, is equivalent to \((\neg p \lor (q \lor \neg r))\) which is in DNF.

2. We start with the truth tables:

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

\[ (q \land p) \quad \neg (q \land p) \quad (q \lor \neg (q \land p)) \quad \neg (q \lor \neg (q \land p)) \quad P \]

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

It follows that \(P\) is not satisfiable.

For the tableau, let \(S_0 = \{P\}\), where \(P = (p \land \neg (q \lor \neg (q \land p)))\). The rule applications are as follows:

- An application of \(\Rightarrow \land\) to \(S_0\) gives
  \[ S_1 = S_0 \cup \{p, \neg (q \lor \neg (q \land p))\} \]

- An application of \(\Rightarrow \lor\) to \(S_1\) gives:
  \[ S_2 = S_1 \cup \{\neg q, \neg (q \land p)\} \]

- An application of \(\Rightarrow \neg\) to \(S_2\) gives:
  \[ S_3 = S_2 \cup \{(q \land p)\} \]

- An application of \(\Rightarrow \land\) to \(S_3\) gives:
  \[ S_4 = S_3 \cup \{q, p\} \]

\(S_4\) contains the clash \(\{q, \neg q\}\). Thus \(P\) is not satisfiable.

3. The translations are as follows:
   - Peter is a lecturer: \(\text{lecturer}(\text{Peter})\).
   - Peter teaches John: \(\text{teaches}(\text{Peter}, \text{John})\).
   - Peter teaches somebody: \(\exists y. \text{teaches}(\text{Peter}, y)\).
   - Somebody teaches John: \(\exists x. \text{teaches}(x, \text{John})\).
   - Everybody who teaches somebody is a lecturer: \(\forall x. (\exists y. \text{teaches}(x, y) \rightarrow \text{lecturer}(x))\)
   - John does not teach anybody: \(\neg \exists x. \text{teaches}(\text{John}, x)\).
4. Let \( S = \{ R, Q, c \} \) be a signature, where \( R \) is a binary predicate symbol, \( Q \) is a unary predicate symbol, and \( c \) is an individual constant. Let
\[
\mathcal{F} = (D^F, R^F, Q^F, c^F)
\]
be the \( S \)-structure defined by
- \( D^F = \{ c, d, e, f \} \);
- \( R^F = \{ (c, d), (d, e), (e, f) \} \);
- \( Q^F = \{ d \} \);
- \( c^F = c \).

(a) \( \mathcal{F} \models Q(c) \) does not hold since \( c^F = c \notin Q^F \).
(b) \( \mathcal{F} \models \exists x. Q(x) \) holds since \( Q^F \neq \emptyset \).
(c) \( \mathcal{F} \models \forall x. \exists y. R(x, y) \) does not hold since there exists no \( g \in D^F \) with \( (f, g) \in R^F \).
(d) \( \mathcal{F} \models \forall x (Q(x) \rightarrow \exists y. R(x, y)) \) holds since \( d \) is the only element of \( Q^F \) and \( (d, e) \in R^F \).
(e) There exists a variable assignment \( a \) such that \( (\mathcal{F}, a) \models R(c, x) \), namely \( a(x) = d \).

5. Let \( S \) be the signature of \( X \) and \( G \). Then \( X \models G \) if the following holds for all \( S \)-structures \( \mathcal{F} \): if \( \mathcal{F} \models H \) for all \( H \in X \), then \( \mathcal{F} \models G \). Students should then discuss various aspects of the two notions. For example, for finite \( \mathcal{F} \), checking \( \mathcal{F} \models G \) corresponds to querying relational databases and can be done very efficiently for small (fixed) \( G \). On the other hand, the problem “\( X \models G \)” is undecidable and investigated in automated theorem proving. They could also mention the distinction between program verification using model checking and deduction.