CSC236, Summer 2004, Assignment 3, sample solution

1. Use structural induction to prove the following claims about propositional formulas.

   (a) Let $PV = \{P_1, \ldots, P_k\}$ be a set of propositional variables, and F_{PV} be the set of all propositional formulas over PV, using Definition 5.1 on page 114 of the Course Notes. For any $e \in F_{PV}$, define $\mathbf{sf}(e)$ as the number of sub-formulas contained in e as substrings (including e itself). Define $\mathbf{pr}(e)$ as the number of parentheses in e, and define $\mathbf{ng}(e)$ as the number of unary connectives $\neg$ contained in e.

   CLAIM: For any $e \in F_{PV}$, $\mathbf{sf}(e) = \mathbf{pr}(e) + \mathbf{ng}(e) + 1$.

   PROOF (STRUCTURAL INDUCTION ON e): Let my predicate $P(e)$ be "$\mathbf{sf}(e) = \mathbf{pr}(e) + \mathbf{ng}(e) + 1$." I will use structural induction to show that $P(e)$ is true for all $e \in F_{PV}$.

   BASIS: Consider the case where $e = P_i$, one of the propositional variables. Then $\mathbf{sf}(e) = 1$ (a propositional variable contains only itself as a sub-formula), and $\mathbf{pr}(e) = \mathbf{ng}(e) = 0$ (a propositional variable contains no parentheses or negations). So

   $$\mathbf{sf}(e) = 1 = 0 + 0 + 1 = \mathbf{pr}(e) + \mathbf{ng}(e) + 1,$$

   in this case, and the basis holds.

   INDUCTION STEP: Assume that e_1 and e_2 are formulas in F_{PV}, and that $P(e_1)$ and $P(e_2)$ hold. There are two cases to consider

   CASE 1: $e = (e_1 * e_2)$, where $* \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$ is a binary connective. In this case we have

   $$\begin{array}{rrcl} [e \text{ adds one subformula:}] & \mathbf{sf}(e) & = & \mathbf{sf}(e_1) + \mathbf{sf}(e_2) + 1 \\ [\text{IH:}] & & = & \mathbf{pr}(e_1) + \mathbf{ng}(e_1) + 1 + \mathbf{pr}(e_2) + \mathbf{ng}(e_2) + 1 + 1 \\ [e \text{ adds two parentheses:}] & & = & \mathbf{pr}(e) + \mathbf{ng}(e_1) + \mathbf{ng}(e_2) + 1 \\ [e \text{ adds zero negations:}] & & = & \mathbf{pr}(e) + \mathbf{ng}(e) + 1, \end{array}$$

   so $P(e_1)$ and $P(e_2)$ imply $P(e)$ in this case.

   CASE 2: $e = \neg e_1$. In this case we have

   $$\begin{array}{rrcl} [e \text{ adds one subformula:}] & \mathbf{sf}(e) & = & \mathbf{sf}(e_1) + 1 \\ [\text{IH:}] & & = & \mathbf{pr}(e_1) + \mathbf{ng}(e_1) + 1 + 1 \\ [e \text{ adds one negation:}] & & = & \mathbf{pr}(e_1) + \mathbf{ng}(e) + 1 \\ [e \text{ adds no parentheses:}] & & = & \mathbf{pr}(e) + \mathbf{ng}(e) + 1, \end{array}$$

   so $P(e_1)$ implies $P(e)$ in this case.

   In either case $P(\{e_1, e_2\})$ imply $P(e)$.

   I conclude that $P(e)$ is true for all $e \in F_{PV}$.

   (b) Define F_{PV}^{+} as the subset of formulas from part (a) that do NOT contain the unary connective $\neg$, and define $\mathbf{sf}(e)$ as in part (a).

   CLAIM: For any $e \in F_{PV}^{+}$, $\mathbf{sf}(e)$ is odd.

PROOF (INDUCTION ON e): Let my predicate $P(e)$ be "$\mathbf{sf}(e)$ is odd." I will use structural induction to show that $P(e)$ is true for all $e \in F_{PV}^{+}$.

BASIS: Consider $e = P_i$, one of the propositional variables. Then $\mathbf{sf}(e) = 1$ (a propositional variable contains only itself as a sub-formula), so the claim holds for the base case.

INDUCTION STEP: Suppose e_1 and e_2 are elements of F_{PV}^{+} and that $P(e_1)$ and $P(e_2)$ both hold. If $e = (e_1 * e_2)$, where $* \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$ is a binary connective, I will show that $P(e)$ also holds. In this case we have

$$\begin{aligned}
\text{[e adds one subformula:]} \quad \mathbf{sf}(e) &= \mathbf{sf}(e_1) + \mathbf{sf}(e_2) + 1 \\
\text{[IH:]} \quad &= 2k + 1 + 2k' + 1 + 1 \quad k, k' \in \mathbb{Z} \\
&= 2(k + k' + 1) + 1, \quad \text{(an odd number),}
\end{aligned}$$

Thus $P(\{e_1, e_2\})$ implies $P(e)$.

I conclude that $P(e)$ holds for every e considered by the basis or induction step. Clearly every such e is an element of F_{PV}^{+}, since it is an element of F_{PV} and has no negations. To see the opposite containment, let e be an arbitrary element of F_{PV}^{+}. If e is a propositional variable, then $P(e)$ holds by the basis step. Otherwise, e is formed using formulas e_1 and e_2 and a binary connective (e has no negations), and $P(e)$ holds by the induction step.

Thus $P(e)$ holds for every $e \in F_{PV}^{+}$. QED.

(c) Define F_{PV} as in part (a), and for any $e \in F_{PV}$ define $\mathbf{par}(e)$ as $\mathbf{sf}(e) \bmod 2$.

CLAIM: For any $e \in F_{PV}$, $\mathbf{par}(e) = (\mathbf{ng}(e) + 1) \bmod 2$.

PROOF (STRUCTURAL INDUCTION ON e): Let my predicate $P(e)$ be "$\mathbf{par}(e) = (\mathbf{ng}(e) + 1) \bmod 2$." I will show this claim is true for all $e \in F_{PV}$ using structural induction.

BASIS: Suppose $e = P_i$, a propositional variable. Then $\mathbf{par}(e) = 1 \bmod 2 = 1$, and $\mathbf{ng}(e) = 0$ (since e, being a propositional variable, has no negations), so $(\mathbf{ng}(e) + 1) \bmod 2 = 1$, and the claim holds for the base case.

INDUCTION STEP: Suppose that e_1 and e_2 are formulas in F_{PV}, and that $P(e_1)$ and $P(e_2)$ hold. There are two cases to consider.

CASE 1: If $e = \neg e_1$, then e has exactly one more sub-formula and one more negation than e_1, and

$$\begin{aligned}
\text{[definition of $\mathbf{par}(e)$:]} \quad \mathbf{par}(e) &= \mathbf{sf}(e) \bmod 2 \\
\text{[e adds one subformula:]} \quad &= (\mathbf{sf}(e_1) + 1) \bmod 2 \\
\text{[definition of $\mathbf{par}(e_1)$, and Fact 1 from A1:]} \quad &= (\mathbf{par}(e_1) + 1 \bmod 2) \bmod 2 \\
\text{[IH:]} \quad &= ((\mathbf{ng}(e_1) + 1) \bmod 2 + 1 \bmod 2) \bmod 2 \\
\text{[e adds one negation:]} \quad &= (\mathbf{ng}(e) \bmod 2 + 1 \bmod 2) \bmod 2 \\
\text{[Fact 1 from A1:]} \quad &= (\mathbf{ng}(e) + 1) \bmod 2
\end{aligned}$$

CASE 2: If $e = (e_1 * e_2)$, where $* \in \{\wedge, \vee, \rightarrow, \leftarrow\}$ is a binary connective, then e has one more sub-formula, and zero more negations than those in e_1 and e_2, so

$$\begin{aligned}
\text{[definition of $\mathbf{par}(e)$:]} \quad \mathbf{par}(e) &= \mathbf{sf}(e) \bmod 2 \\
\text{[e adds one subformula:]} \quad &= (\mathbf{sf}(e_1) + \mathbf{sf}(e_2) + 1) \bmod 2 \\
\text{[definition of $\mathbf{par}(e)$ and Fact 1 twice:]} \quad &= (\mathbf{par}(e_1) + \mathbf{par}(e_2) + 1 \bmod 2) \bmod 2 \\
\text{[IH:]} \quad &= ((\mathbf{ng}(e_1) + 1) \bmod 2 + (\mathbf{ng}(e_2) + 1) \bmod 2 \\
&\qquad + 1 \bmod 2) \bmod 2 \\
\text{[Fact 1 twice:]} \quad &= ((\mathbf{ng}(e_1) + \mathbf{ng}(e_2) + 2) \bmod 2 \\
&\qquad + 1 \bmod 2) \bmod 2 \\
\text{[e adds zero negations, Fact 1:]} \quad &= (\mathbf{ng}(e) + 3) \bmod 2 \\
\text{[Fact 1 twice:]} \quad &= (\mathbf{ng}(e) + 1) \bmod 2
\end{aligned}$$

In both cases, $P(\{e_1, e_2\})$ imply $P(e)$.
I conclude that $P(e)$ is true for all $e \in F_{PV}$. QED.

2. In the propositional formulas below I use the rules of precedence from sections 5.2.4 and 5.6 of the Course Notes to reduce the number of parentheses. In your solution you are welcome to re-introduce parentheses if it makes things clearer.

   (a) Use the logical equivalences in section 5.6 (no truth tables) to prove that

   $$P \to Q \to R \to S \quad \text{LEQV} \quad \neg(P \wedge Q \wedge R \wedge \neg S).$$

   SOLUTION:

   $$\begin{array}{rrl} \text{[insert parentheses:]} \quad P \to Q \to R \to S & \text{LEQV} & (P \to (Q \to (R \to S))) \\ [\to \text{ rule three times:}] & \text{LEQV} & (\neg P \vee (\neg Q \vee (\neg R \vee S))) \\ \text{[De Morgan's law three times:]} & \text{LEQV} & \neg(P \wedge (Q \wedge (R \wedge \neg S))) \\ \text{[Reduce parentheses with precedence rules:]} & \text{LEQV} & \neg(P \wedge Q \wedge R \wedge \neg S) \end{array}$$

   (b) Write a CNF formula equivalent to both formulas in part (a).

   SOLUTION: Consider the formula

   $$\neg P \vee \neg Q \vee \neg R \vee S,$$

   a disjunction of literals (a MAXTERM). A conjunction of maxterms is a CNF formula, so a single maxterm is a CNF formula (and, simultaneously, a DNF formula). The given formula is equivalent to the formulas in (a), since it is the equivalent formula on the second line (minus the parentheses).

   (c) Use the logical equivalences in section 5.6 (no truth tables) to prove that

   $$P \leftrightarrow Q \leftrightarrow R \quad \text{LEQV} \quad P \wedge Q \wedge R \vee P \wedge \neg Q \wedge \neg R \vee \neg P \wedge \neg Q \wedge R \vee \neg P \wedge Q \wedge \neg R.$$

   SOLUTION:

   $$\begin{array}{rrl} P \leftrightarrow Q \leftrightarrow R & \text{LEQV} & (P \leftrightarrow (Q \leftrightarrow R)) \\ [\leftrightarrow \text{ rule twice:}] & \text{LEQV} & ((P \wedge (Q \wedge R \vee \neg Q \wedge \neg R)) \\ & & \quad \vee (\neg P \wedge \neg(Q \wedge R \vee \neg Q \wedge \neg R))) \\ \text{[Distributivity, De Morgan:]} & \text{LEQV} & P \wedge Q \wedge R \vee P \wedge \neg Q \wedge \neg R \\ & & \quad \vee \neg P \wedge ((\neg Q \vee \neg R) \wedge (Q \vee R)) \\ \text{[Distributivity twice:]} & \text{LEQV} & P \wedge Q \wedge R \vee P \wedge \neg Q \wedge \neg R \\ & & \quad \vee \neg P \wedge ((\neg Q \wedge Q \vee \neg Q \wedge R) \vee (\neg R \wedge Q \vee \neg R \wedge R)) \\ \text{[Commutativity, Identity twice:]} & \text{LEQV} & P \wedge Q \wedge R \vee P \wedge \neg Q \wedge \neg R \\ & & \quad \vee \neg P \wedge (\neg Q \wedge R \vee \neg R \wedge Q) \\ \text{[Distributivity twice:]} & \text{LEQV} & P \wedge Q \wedge R \vee P \wedge \neg Q \wedge \neg R \\ & & \quad \vee \neg P \wedge \neg Q \wedge R \vee \neg P \wedge \neg R \wedge Q \\ \text{[Commutativity:]} & \text{LEQV} & P \wedge Q \wedge R \vee P \wedge \neg Q \wedge \neg R \\ & & \quad \vee \neg P \wedge \neg Q \wedge R \vee \neg P \wedge Q \wedge \neg R \end{array}$$

   (d) Use the logical equivalences from section 5.6, and part (c) of this assignment, to prove that $\leftrightarrow$ is associative, in other words prove that

   $$(P \leftrightarrow (Q \leftrightarrow R)) \quad \text{LEQV} \quad ((P \leftrightarrow Q) \leftrightarrow R)$$

SOLUTION: The formula in part (c) is symmetrical with respect to P, Q, and R, so use commutativity and run the movie backwards:

$$\begin{array}{rll}
\text{[by part (c):]}\ (P \leftrightarrow (Q \leftrightarrow R)) & \text{LEQV} & P \wedge Q \wedge R \vee P \wedge \neg Q \wedge \neg R \\
 & & \quad \vee \neg P \wedge \neg Q \wedge R \vee \neg P \wedge Q \wedge \neg R \\
\text{[Associativity, Commutativity:]} & \text{LEQV} & P \wedge Q \wedge R \vee \neg P \wedge \neg Q \wedge R \\
 & & \quad \vee \neg P \wedge Q \wedge \neg R \vee \neg Q \wedge P \wedge \neg R \\
\text{[Distributivity:]} & \text{LEQV} & (P \wedge Q \vee \neg P \wedge \neg Q) \wedge R \\
 & & \quad \vee (\neg P \wedge Q \vee \neg Q \wedge P) \wedge \neg R \\
\text{[}\leftrightarrow\text{ law, identity, commutativity:]} & \text{LEQV} & (P \leftrightarrow Q) \wedge R \\
 & & \quad \vee (\neg P \wedge P \vee \neg P \wedge Q \vee \neg Q \wedge P \vee \neg Q \wedge Q) \wedge \neg R \\
\text{[Distributivity]} & \text{LEQV} & (P \leftrightarrow Q) \wedge R \\
 & & \quad \vee ((\neg P \vee \neg Q) \wedge (P \vee Q)) \wedge \neg R \\
\text{[De Morgan's law twice:]} & \text{LEQV} & (P \leftrightarrow Q) \wedge R \\
 & & \quad \vee \neg(P \wedge Q \vee \neg P \wedge \neg Q) \wedge \neg R \\
\text{[}\leftrightarrow\text{ law twice:]} & \text{LEQV} & ((P \leftrightarrow Q) \leftrightarrow R)
\end{array}$$

3. Consider the truth table for $(P \oplus Q)$, on page 141 of the Course notes.

   (a) Write the truth table for $(P \oplus (Q \oplus R))$.

   SOLUTION: I write all possible combinations of truth assignments to propositional variables P, Q, and R, and use the truth table on page 141 to extend these to the given formula. I use the final columns to match minterms and maxterms to truth assignments that (respectively) satisfy or falsify the formula.

P	Q	R	$(Q \oplus R)$	$(P \oplus (Q \oplus R))$	MINTERMS	MAXTERMS
0	0	0	0	0		$P \vee Q \vee R$
0	0	1	1	1	$\neg P \wedge \neg Q \wedge R$	
0	1	0	1	1	$\neg P \wedge Q \wedge \neg R$	
0	1	1	0	0		$P \vee \neg Q \vee \neg R$
1	0	0	0	1	$P \wedge \neg Q \wedge \neg R$	
1	0	1	1	0		$\neg P \vee Q \vee \neg R$
1	1	0	1	0		$\neg P \vee \neg Q \vee R$
1	1	1	0	1	$P \wedge Q \wedge R$	

   (b) Write a CNF formula equivalent to $(P \oplus (Q \oplus R))$.

   SOLUTION: The conjunction of the maxterms in the last column of the table from the previous question will be false exactly when our formula is false, hence it is an equivalent CNF formula:

$$(P \vee Q \vee R) \wedge (P \vee \neg Q \vee \neg R) \wedge (\neg P \vee Q \vee \neg R) \wedge (\neg P \vee \neg Q \vee R)$$

   (c) Write a DNF formula equivalent to $(P \oplus (Q \oplus R))$

   SOLUTION: The disjunction of minterms from the second-last column of part (a) will be true exactly when our formula is true, hence it is an equivalent DNF:

$$\neg P \wedge \neg Q \wedge R \vee \neg P \wedge Q \wedge \neg R \vee P \wedge \neg Q \wedge \neg R \vee P \wedge Q \wedge R.$$

.

   (d) Is $\oplus$ associative? In other words, is the formula

$$((P \oplus (Q \oplus R)) \leftrightarrow ((P \oplus Q) \oplus R))$$

   a tautology? Prove your claim.

CLAIM: The formula is a tautology.

PROOF: Consider the 8 possible truth assignments to propositional variables P, Q, and R, and then use the truth table from page 141 to find extended truth assignments:

P	Q	R	$(P \oplus Q)$	$(Q \oplus R)$	$(P \oplus (Q \oplus R))$	$((P \oplus Q) \oplus R)$
0	0	0	0	0	0	0
0	0	1	0	1	1	1
0	1	0	1	1	1	1
0	1	1	1	0	0	0
1	0	0	1	0	1	1
1	0	1	1	1	0	0
1	1	0	0	1	0	0
1	1	1	0	0	1	1

Since each of the possible truth assignments either satisfies both $(P \oplus (Q \oplus R))$ and $((P \oplus Q) \oplus R)$, or else falsifies them both, they are logically equivalent formulas, and (by definition of logical equivalence) the formula

$$(P \oplus (Q \oplus R)) \leftrightarrow ((P \oplus Q) \oplus R)$$

is a tautology. QED.

4. Let $\mathcal{LA}^*$ be a first-order language with predicates L (of arity 2), $\approx$ (of arity 2, the equality predicate), and W (of arity 1), no constant symbols, and an infinite set of variables that include x, y, z, w. Consider the formula

$$F: \qquad \exists z W(z) \rightarrow \exists y \forall x (W(y) \wedge (W(x) \rightarrow (L(y,x) \vee \approx (y,x))))$$

(a) Does every interpretation of $\mathcal{LA}^*$ that has domain $D = \mathbb{N}$, and $L(z,w)$ interpreted as $z < w$ satisfy F? Explain your answer. (Notice that you must consider all possible interpretations of predicate W).

SOLUTION: Yes. The only way to falsify the formula is if the antecedent is satisfied and the consequent is falsified.

Suppose the antecedent is satisfied. The predicate W corresponds to a subset of $\mathbb{N}$, $W^S = \{x \in \mathbb{N} : W(x) = \text{true}\}$, so $\exists z W(z)$ corresponds to W^S being non-empty. Every non-empty subset of $\mathbb{N}$ has a least element, so let the least element of W^S be v_W.

Now the consequent is satisfied: there is a y (namely v_W) such that for every x you have $y \in W^S$ (by choice of y) and if $x \in W^S$, then either $y < x$ or $y = x$ (by choice of y to be the least element).

(b) Does every interpretation of $\mathcal{LA}^*$ that has domain $D = \mathbb{R}$ (the real numbers), and $L(z,w)$ interpreted as $z < w$ satisfy F? Explain your answer.

SOLUTION: No. Let the predicate W correspond to the non-empty subset of the real numbers $W^S = (0,1)$ (the open interval from 0 to 1, exclusive). This interval has no least element, so for every valuation of y there is a valuation of x such that $x < y$, which falsifies both $L(y,x)$ and $\approx (y,x)$. This is a counterexample, under this interpretation the antecedent is satisfied by the consequent is falsified.

(c) Modify the formula by switching quantifiers:

$$F': \qquad \exists z W(z) \rightarrow \forall x \exists y (W(y) \wedge (W(x) \rightarrow (L(y,x) \vee \approx (y,x)))).$$

Does every interpretation of $\mathcal{LA}^*$ that has domain $\mathbb{R}$ and $L(z,w)$ interpreted as $z < w$ satisfy F'? Explain your answer.

SOLUTION: Yes. Assume you have an arbitrary interpretation where the antecedent is satisfied (otherwise the entire formula is vacuously satisfied). Then W^S is a non-empty subset of the real numbers. Given any choice of x, there are two possibilities. If $x \in W^S$, then choose $y = x$, and for sub-formula quantified by $\forall x \exists y$ is satisfied. If $x \notin W^S$, let y be any element of W^S, and the sub-formula quantified by $\forall x \exists y$ is satisfied (because the antecedent $W(x)$ is false). Thus a suitable y can always be chosen, and the formula as a whole is always satisfied.

(d) What CSC236 concept does the original formula F express?

SOLUTION: The formula expresses the Principle of Well-Ordering for an arbitrary domain D.

5. Let $\mathcal{LF}^+$ be the language of family relations with an infinite set of variables including x, y, and z, predicate symbols P (of arity 3), S (of arity 2), and $\approx$ (of arity 2, the equality predicate). Consider an interpretation where D is the set of all human beings, $P(x, y, z)$ means x and y are the parents of z, $S(x, y)$ means x is a sibling of y. State in careful English what it means for x to be a cousin of y. Then write a formula in $\mathcal{LF}^+$ that means that x is a cousin of y.

SOLUTION: The main thing is to make sure your English definition of cousin matches your predicate logic definition. Your definitions don't have to match mine (or anybody else's), but they should match each other.

DEFINITION: Two individuals, A and B, are cousins if one of A's parents is a sibling of one of B's parents, and no parent of A is also a parent of B.

REMARK: In this definition, if my brother and I each had offspring by the same mother, the offspring would be half-siblings, and a little too close for cousins.

FORMULA:

$$\mathbf{C}(x, y): \quad \exists u \exists v \exists t \exists s (P(u, t, x) \wedge P(v, s, y) \wedge S(u, v)) \wedge \forall p \forall q \forall r (P(p, q, x) \rightarrow \neg P(p, r, y))$$

REMARK: The predicate $S(x, y)$ is symmetrical in its arguments, and $P(x, y, z)$ is symmetrical in its first two arguments. This saves listing all possible permutations of the first two arguments in the above case.