Homework 2 (due April 15 2020)

Directions: *Write up carefully argued solutions to the following problems. Your solution should be clear enough to convince someone who does not already know the answer. You may use results from lecture and previous homeworks without proof. See the syllabus for more details and for permitted resources and collaboration.*

For proofs in Propositional Logic, below, you must cite every rule that you apply, including Commutativity *and* Associativity*. You may apply only a single rule per line. However, you can apply that rule to multiple parts of the formula as long as they are non-overlapping.*

1. Equivalence proofs (20 points)

Prove the following assertions using logical equivalences.

(a) [6 points] $p \wedge (p \rightarrow q) \equiv p \wedge q$.

(b) [6 points] $((p \wedge q) \vee (p \rightarrow (\neg p \wedge r))) \vee p \equiv \mathsf{T}$.

(c) [8 points] $p \rightarrow (q \rightarrow r) \equiv q \rightarrow (p \rightarrow r)$.

2. Two of a kind (20 points)

(a) [2 points] Translate the Boolean Algebra expression $((X \cdot (X+Y))' + X \cdot (Y + (X + Y')))'$ to Propositional Logic. Use the variables p and q to represent the propositions $X = 1$ and $Y = 1$, respectively.

(b) [16 points] Prove that your solution to (a) is a contradiction using a chain of equivalences.

(c) [2 points] Why do we know that the Boolean Algebra expression from part (a) is always 0? Explain.

3. Counting courses with combinational circuts (16 points)

In Lecture 4, we constructed a circuit to compute the number of classes (lectures or sections) remaining on a given day of the week.

(a) [5 points] Write c_0 in the product of sum form.

(b) [8 points] Simplify the sum of product forms of c_1 using boolean algebra axioms and theorems. Make sure to cite which axioms and theorems you are using when simplifying.

(c) [3 points] Write down the boolean algebra expressions, with c_0 and c_1 as the variables, that are true exactly when the output of the `classesLeft` function is

1. an integer of the form 2^n (that is, a power of 2),
2. an integer of the form $2^n - 1$ (that is, the predecessor of a power of 2),
3. an integer less than 3.

4. From predicate logic to English (16 points)

Let the domain of discourse be people in France. Let's define the predicates $\mathsf{Francophone}(x)$ and $\mathsf{Anglophone}(x)$ to mean that x is a French speaker or an English speaker, respectively. Define the predicates $\mathsf{French}(x)$, $\mathsf{Parisian}(x)$, and $\mathsf{Tourist}(x)$ to mean that x is French, lives in Paris, or is a tourist, respectively.

Translate each of the following logical statements into English. You should not simplify. However, you should use the techniques shown in lecture for producing more natural translations when restricting domains and for avoiding the introduction of variable names when not necessary.

(a) [4 points] $\forall x.(\mathsf{Parisian}(x) \rightarrow (\mathsf{French}(x) \vee \mathsf{Tourist}(x)))$

(b) [4 points] $\exists x.(\mathsf{Anglophone}(x) \wedge \mathsf{Tourist}(x) \wedge \neg\, \mathsf{Francophone}(x))$

(c) [4 points] $\Big(\forall x.\big((\mathsf{Parisian}(x) \land \neg\,\mathsf{Tourist}(x)) \to \mathsf{French}(x)\big)\Big) \land \neg\Big(\forall x.\big(\,\mathsf{French}(x) \to (\mathsf{Parisian}(x) \lor \mathsf{Tourist}(x))\big)\Big)$

(d) [4 points] $\neg\exists x.\Big(\big(\,\mathsf{Tourist}(x) \land \neg(\mathsf{Francophone}(x) \lor \mathsf{Anglophone}(x))\big) \land \mathsf{Parisian}(x)\Big)$

5. From English to predicate logic (16 points)

Let the domain of discourse be all procedures. We define the predicates $\mathsf{Caller}(x)$ to mean that x is a procedure that calls another procedure as a subroutine, and $\mathsf{Callee}(x)$ to mean that x is called by another procedure as a subroutine. We also define the predicate $\mathsf{Calls}(x, y)$ to mean that the procedure x calls the procedure y as a subroutine. You can also assume an "=" operator that is true when x and y are the same procedure.

Translate each of the following English statements into predicate logic. Do not simplify.

(a) [4 points] Not every procedure is a caller or a callee.

(b) [4 points] The merge procedure has one caller. (Use the constant Merge to refer to the merge procedure.)

(c) [4 points] Not every callee has only one caller.

(d) [4 points] Some callee has more than one caller.

6. Beyond compare (12 points)

The questions below consider the two propositions

$$\exists x.(P(x) \land Q(x)) \quad \text{and} \quad (\exists x.P(x)) \land (\exists x.Q(x))$$

where P and Q are predicates.

(a) [6 points] Give examples of predicates P and Q and a domain of discourse so that the two propositions are **not** equivalent.

(b) [6 points] Give examples of predicates P and Q and a domain of discourse so that they **are** equivalent.

(c) [0 points] **Extra credit**: What logical relationship holds between these two propositions? Explain.