CSE 311: Foundations of Computing I

Homework 2 (due Wednesday, April 17th at 11:00 PM)

Directions: *Write up carefully argued solutions to the following problems. Your solution should be clear enough that it should explain to someone who does not already understand the answer why it works. However, you may use results from lecture, the theorems handout, and previous homeworks without proof.*

Below, for proofs in Propositional Logic, you must cite every rule that you apply, including *commutativity* and *associativity*. Apply only a single equivalence at a time, and state its name where used.

For Boolean Algebra, however, you may skip citing *associativity* throughout your answer. In fact, you may skip the parentheses altogether when their only purpose is to indicate the order of application for two identical operators, e.g., $(X + Y) + Z$ vs $X + (Y + Z)$ or $(XY)Z$ vs $X(YZ)$.[1]

1. All the Things You Are (24 points)

Prove the following assertions using logical equivalences.

(a) [6 Points] $p \to (q \to r) \equiv q \to (p \to r)$.

(b) [6 Points] $q \to (p \to r) \equiv (p \land q) \to r$.

(c) [12 Points] $(\neg p \to q) \land (p \to q) \equiv q$. We will call this fact "**Proof by Cases**".

2. Straight No Chaser (10 points)

Use the theorems and axioms of Boolean algebra to prove that $X' + XY' + Z' + ZY$ is a tautology.

3. It Never Entered My Mind (22 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. The following problems refer to the output bits of that circuit: c_0, c_1, c_2, c_3.

(a) [6 Points] Write c_1 in the product-of-sums form.

(b) [6 Points] Write c_0 in the product-of-sums form.

(c) [10 Points] The sum-of-products form of c_1 was given in lecture. Show how to simplify it using the theorems and axioms of Boolean algebra.

4. Take the A Train (16 points)

Let the domain of discourse be trains. Let's define the predicates Diesel(x) and Electric(x) to mean that x is a diesel-driven or electric-driven train, respectively. Define the predicates Passenger(x), Cargo(x), and Mail(x) to mean that x carries passengers, cargo or mail, 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 really necessary.

(a) [4 Points] $\forall x\ ((\text{Diesel}(x) \land \neg\,\text{Electric}(x)) \lor (\neg\,\text{Diesel}(x) \land \text{Electric}(x)))$

[1] Indeed, readers of mathematics will expect "+" and "×" to be operations that are associative, as in regular algebra, so it would be confusing (poor style) to use them to refer to operations that are not associative.

(b) [4 Points] $\exists x\,(\text{Cargo}(x) \wedge \neg\,\text{Passenger}(x) \wedge \text{Mail}(x))$

(c) [4 Points] $\forall x\,((\text{Diesel}(x) \wedge \text{Cargo}(x)) \rightarrow \neg\,\text{Passenger}(x))$

(d) [4 Points] $\neg\exists x\,(\neg(\text{Passenger}(x) \wedge \text{Mail}(x)) \wedge \neg\,\text{Cargo}(x))$

5. It Don't Mean a Thing (16 points)

Let the domain of discourse be all people in the CSE department. Let's define the predicates Student(x) and Instructor(x) to mean that x is a student or an instructor, respectively. Define the predicate TakesCourseFrom(x, y) to mean that x is a student taking a course from instructor y. You may also assume the existence of an operator "$=$" such that $x = y$ is true if and only if x and y are the same person.

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

(a) [4 Points] Everyone who takes some course is a student.

(b) [4 Points] Jessica is both a student and an instructor, but she does not take a course from anyone who takes a course from her. (Use the constant "Jessica" to refer to Jessica in your logical statement.)

(c) [4 Points] Some students take courses from more than one instructor.

(d) [4 Points] There is a student that has exactly one instructor.

6. Compared To What (12 points)

The questions below consider the two propositions

$$\forall x\,(P(x) \vee Q(x)) \quad \text{and} \quad (\forall x\,P(x)) \vee (\forall x\,Q(x))$$

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.

7. Extra credit: At Last (0 points)

In this problem, you will design a circuit with a minimal number of gates that takes a pair of four-bit integers $(x_3x_2x_1x_0)_2$ and $(y_3y_2y_1y_0)_2$ and returns a single bit indicating whether $x_3x_2x_1x_0 < y_3y_2y_1y_0$. See the following table for some examples.

$x_3x_2x_1x_0$	$y_3y_2y_1y_0$	$x_3x_2x_1x_0 < y_3y_2y_1y_0$
0101	1011	1
1100	0111	0
1101	1101	0

Design such a circuit using at most 10 AND, OR, and XOR gates. You can use an arbitrary number of NOT gates, and a single gate can have multiple inputs. (Extra credit points start at 10 gates, but if you can use fewer, you will get even more points.)