CSE 311: Foundations of Computing (Autumn, 2015)

Homework 2 Out: Friday, 9-Oct. Due: Friday, 16-Oct, **11:59PM** on **Gradescope**

1. Logical equivalence [20 points]

Show that the following expressions are tautologies. You should use only the equivalences in the "Properties of logical equivalences" handout on the web page. You don't need to label the equivalences by name, but you should use only one equivalence per step, except for associativity and commutativity of ∧ and ∨ which you can use multiple times in the same line.

(a) $(p \leftrightarrow q) \leftrightarrow \big((\neg p \vee q) \wedge (\neg q \vee p)\big)$

(b) $\big((p \rightarrow q) \wedge (r \rightarrow \neg q)\big) \rightarrow (r \rightarrow \neg p)$

2. Contrapositing [12 points]

State in English the contrapositive of each of the following implications. Be sure to use de Morgan's law or the law of implication (or both) to simplify the statements so they read more naturally in English.

(a) If logic is consistent and prime numbers do not exist, then any positive integer divides 2.

(b) If every integer greater than 7 is prime, then every integer less than 10 is even.

3. Bit counter [25 points]

Given a three-bit integer $x_0x_1x_2$, you will design a logical circuit that counts the number of bits of $x_0x_1x_2$ which are equal to 1. The output has two bits y_0, y_1 where the number y_0y_1 is the binary representation of the number of input bits that are equal to 1. For example, if $x_0x_1x_2 = 111$ then the output is $y_0y_1 = 11$, and if $x_0x_1x_2 = 101$, then the output is $y_0y_1 = 10$.

(a) Write the truth table of the output, the sum-of-product and the product-of-sum forms for each of the output bits.

(b) Use the laws of Boolean algebra to derive simplified expression for y_0 and for y_1 (you can start with either the sum-of-products or product-of-sums expression).

(c) Implement y_0 and y_1 with a digital circuit (using your simplified expressions). You are only allowed to use NOR gates in your design.

4. Domain of discourse [10 points]

(a) Give examples of predicates P and Q and a domain of discourse so that the two statements

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

are not equivalent.

(b) Give an example where they **are** equivalent.

(c) [mini extra credit]: Say that a logical connective $p \otimes q$ is **non-trivial** if it sometimes evaluates to false and sometimes evaluates to true. Is there any such connective $p \otimes q$ such that $\exists x\,(P(x) \otimes Q(x))$ and $\exists x\, P(x) \otimes \exists x\, Q(x)$ are logically equivalent for every domain and choice of predicates? Explain.

5. Translating into logic [15 points]

Translate the following sentences into predicate logic. The domain of discourse is the set of all positive integers. You can use the predicates: $\mathrm{Even}(x), \mathrm{Odd}(x), \mathrm{Prime}(x), \mathrm{Greater}(x, y), \mathrm{Equal}(x, y)$, and $\mathrm{Sum}(x, y, z)$ from Lecture 5.

(a) For all positive integers x, there exist positive integers y and z such that $x = y + z$.

(b) For all positive integers x, if x is a prime, there exist primes y and z such that $x = y + z$.

(c) For all positive integers x, there exist prime integers y and z such that $y > x$ and $z > x$ and $z = y + 2$.

(d) There exists a positive integer x that is a prime such that for all positive integers y, if y is even, then $y \geq x$.

(e) There is no positive integer x such that x is even and also a prime.

Extra credit 1: Comparison circuit

In this problem, you will design a circuit with a minimal number of gates that takes a pair of four-bit integers $(x_0x_1x_2x_3)_2$ and $(y_0y_1y_2y_3)_2$ and returns a single bit indicating whether $x_0x_1x_2x_3 < y_0y_1y_2y_3$. See the following table for some examples.

$x_0x_1x_2x_3$	$y_0y_1y_2y_3$	$x_0x_1x_2x_3 < y_0y_1y_2y_3$
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'll get even more points.)

Extra credit 2: Get these MF Pokémon off this MF private jet.

Consider the predicates $\mathrm{Pokemon}(x), \mathrm{Celebrity}(x), \mathrm{Jet}(x), \mathrm{WearsPants}(x)$, $\mathrm{Flies}(x, y)$ which denote, respectively, that x is a Pokémon, celebrity, or jet, that x wears pants, and that person x flies on jet y. We also use $\mathrm{Alive}(x)$ to denote that x is alive. (For the purposes of this problem, celebrities are alive, even if they're a little dead inside. Pokémon are definitely alive.) $\mathcal{T}$ is Taylor's jet and is $\mathcal{B}$ Bieber's jet. The domain of discourse is the set of all alive things and all flying things.

Suppose the following are true:

(i) $\forall x \left(\mathrm{Flies}(x, \boldsymbol{\mathcal{T}}) \leftrightarrow \left(\mathrm{Pokemon}(x) \wedge \mathrm{WearsPants}(x)\right)\right)$

(ii) $\forall x \left(\mathrm{Flies}(x, \boldsymbol{\mathcal{B}}) \leftrightarrow \left(\mathrm{Celebrity}(x) \wedge \mathrm{WearsPants}(x)\right)\right)$

(iii) $\forall j \left(\mathrm{Jet}(j) \rightarrow \exists x\left(\mathrm{Alive}(x) \wedge \neg \mathrm{Flies}(x, j)\right)\right)$

(iv) $\forall j \left(\mathrm{Jet}(j) \rightarrow \exists h \left(\mathrm{Jet}(h) \wedge \forall x \left(\mathrm{Alive}(x) \rightarrow \left(\mathrm{Flies}(x, h) \leftrightarrow \neg\, \mathrm{Flies}(x, j)\right)\right)\right)\right)$

Explain precisely why there is at least one Pokémon who is not allowed to fly on Taylor's jet.