CSE 311: Foundations of Computing I

Homework 2 (due Monday, October 17th at 11:00 PM)

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

1. Express Yourself (8 points)

Consider the following boolean function C:

p	q	r	s	$C(p, q, r, s)$
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

(a) [4 Points] Write a Boolean algebra expression for C in sum-of-products form.

(b) [4 Points] Write a Boolean algebra expression for C in products-of-sums form.

2. Keep It Simple (16 points)

(a) [10 Points] Use Boolean algebra identities to simplify your expression from Problem 1(a) down to an expression that includes *only 3 gates* (each of which is either AND, OR, or NOT).

You should format your work like an equivalence chain with one expression per line and with the name of the identity applied to produce that line written next to it. However, since we are using Boolean algebra notation, which does not include unnecessary parentheses, you should not include lines that apply Associativity or Commutativity. You can also skip lines that simply apply the Identity rule.

Hint: As we saw in lecture, you may need to temporarily *expand* your expression in order to shrink it fully.

(b) [4 Points] Write a truth table for your simplified expression from part (a) and confirm that it matches the one used to define C in Problem 1. As always, be sure to include all subexpressions as their own columns.

(c) [2 Points] Draw your simplified expression from part (a) as a circuit.

3. Pet Project (16 points)

Let the domain of discourse be people and pets. Define the predicates $\textsf{Person}(x)$ to mean that x is a person and $\textsf{Pet}(y)$ to mean that y is a pet. Define the predicate $\textsf{TakesCareOf}(x, y)$ to mean that x takes care of y and the predicate $\textsf{HasPaws}(y)$ to mean that y has paws.

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] $\neg \exists x \, (\textsf{Person}(x) \land \textsf{HasPaws}(x))$

(b) [4 Points] $\exists x \, \big(\textsf{Person}(x) \land \exists y \, (\textsf{Pet}(y) \land \textsf{TakesCareOf}(x, y) \land \textsf{TakesCareOf}(y, x))\big)$

(c) [4 Points] $\exists x \, \big(\textsf{Person}(x) \land \exists y \, (\textsf{Pet}(y) \land \textsf{TakesCareOf}(x, y) \land \neg \, \textsf{HasPaws}(y))\big)$

(d) [4 Points] $\forall x \, \big(\textsf{Pet}(x) \to \exists y \, ((\textsf{Person}(y) \lor \textsf{Pet}(y)) \land \textsf{TakesCareOf}(y, x))\big)$

4. Almost Doesn't Count (20 points)

Let the domain of discourse be people and email messages. Define the predicate Person(x) to mean that x is a person and the predicate Message(y) to mean that y is an email. Also define the predicate Sender(x, y) to mean that x sent message y and the predicate Receiver(x, y) to mean that x received message y.

Let us also assume an "$=$" operator that is true when x and y are the same person or the same email message. In this problem we will use equality to count (all the way to 2!) in predicate logic, without actually using any numbers in our formulas!

Computer scientists count starting from zero. The formula "$\neg\exists y$ Message(y)" can be translated as "there are no messages" or equivalently, "there are zero messages." Hurray, we counted to zero!

(a) [4 Points] Use your new-found ability to count to zero in logic to translate the following English sentence.

Some person has sent zero messages.

Use the predicates from the beginning of this question.

Notice that "Some person..." could also be equivalently phrased as "At least one person...", so actually we also know how to count to (at least) one.

(b) [4 Points] Translate this sentence:

Zero people have sent at least one message.

(c) [4 Points] Let's try counting to two, with the following formula.

$$\exists x \left(\text{Message}(x) \land \exists y \ \text{Message}(y) \right)$$

This formula fails to count to two: it can be true even when there is only one message. Explain why.

(d) [4 Points] We can fix the formula from the previous part by adding $x \neq y$, like this:

$$\exists x \Big(\text{Message}(x) \land \exists y \left(\text{Message}(y) \land x \neq y \right) \Big)$$

We can translate this formula back to English as "There are at least two (different) messages." Use a similar idea to translate the following English sentence into logic:

Someone has received at least two (different) messages.

(e) [4 Points] If we negate the sentence "there are at least two (different) messages", intuitively we should get the sentence "there is at most one message". Show how this works in logic by negating the formula from part (d), copied here for your convenience:

$$\exists x \Big(\text{Message}(x) \land \exists y \left(\text{Message}(y) \land x \neq y \right) \Big)$$

Simplify your answer so that it contains no negations. As always, show your work.

Hint: Try to find opportunities to use implication.

5. Mind Your Ps and Qs (12 points)

Let P and Q be predicates.

(a) [2 Points] Translate the proposition

$$\forall x\,(P(x) \to Q(x))$$

directly into English. This time, *do not* try to make your translation natural sounding. Just do the most literal translation possible.

(b) [2 Points] Translate the proposition

$$(\forall x\,P(x)) \to (\forall x\,Q(x))$$

directly into English. Again, *do not* try to make your translation natural sounding. Just do the most literal translation possible.

(c) [4 Points] Give an example of predicates P and Q and a domain of discourse so that the propositions from parts (a) and (b) **do not** have the same truth value (i.e., one is false and one is true).

(d) [4 Points] Give an example of predicates P and Q and a domain of discourse where the propositions from parts (a) and (b) **do** have the same truth value (i.e., both are false or both are true).

6. What Are You Implying? (16 points)

Consider the following proposition

$$\exists x\,(P(x) \to Q(x))$$

(a) [3 Points] Suppose that $P(x)$ is a *tautology*. Give a simplified expression, **not involving** P, that is equivalent to the one above in this case.

(b) [3 Points] Suppose that $P(x)$ is a *contradiction*. Give a simplified expression, **not involving** P, Q, or $\exists$, that is equivalent to the original proposition.

What can we say about the value of the proposition in this case?

(c) [3 Points] Suppose that $P(x)$ is a *contingency*. Give a simplified expression, **not involving** P, Q, or $\exists$, that is equivalent to the original proposition.

(d) [4 Points] Using what you learned in (a-c), give a simplified expression, **not involving** $\to$, that is equivalent to the original proposition.

(e) [1 Point] Rewrite your simplified expression from (d) as an implication, where the $\to$ is **no longer** inside of a quantifier (as it was originally).

(f) [2 Points] Of the three equivalent propositions — the original one and your answers to (d) and (e) — which do you think is the easiest to understand and which is the hardest to understand?

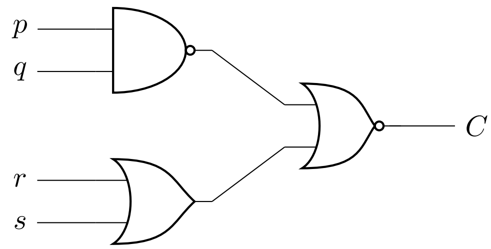
In general, what can we say, in terms of *effective communication*, about propositions that include a $\to$ inside of an $\exists$, as in the original proposition.

7. Can't Get No Satisfaction (12 points)

Suppose that we wish to find a circuit that calculates the following proposition, which we will call W:

$$(\neg p \vee \neg q \vee \neg t) \wedge (p \vee t) \wedge (q \vee t) \wedge (r \vee s \vee \neg u) \wedge (\neg r \vee u) \wedge$$
$$(\neg s \vee u) \wedge (t \vee u \vee v) \wedge (\neg t \vee \neg v) \wedge (\neg u \vee \neg v)$$

Our friend tells us that the circuit below always calculates the same value as W. (This is surprising, since our friend's circuit does not even use the inputs t, u, or v!)



We would like to check that our friend is telling the truth. One way to do so would be to use a truth table, but since the proposition has 7 variables, a truth table would have 128 rows. That is probably too many to do on paper, but no problem at all for software. This problem guides you through how you could use a Satisfiability solver (defined below) to answer this question automatically.

(a) [1 Point] First, translate our friend's circuit into a proposition using only $\neg$, $\wedge$, and $\vee$. Do not simplify your answer.

(b) [2 Points] Write a proposition that is *always true* iff the circuit above correctly calculates the value of W. (*Hint*: Use your answer to part (a).) Explain your answer. Again, do not simplify.

(c) [1 Point] Using either of the techniques shown in lecture, prove the following equivalence:

$$p \leftrightarrow q \equiv \neg p \leftrightarrow \neg q$$

(d) [2 Points] Explain why part (c) tells us that $A \equiv B$ holds iff $\neg A \equiv \neg B$ holds.

(e) [2 Points] Explain why part (d) tells us that $A \equiv B$ holds iff $\neg(A \leftrightarrow B)$ is a *contradiction*.

(f) [2 Points] Write a proposition that is *always false* (i.e., a contradiction) iff the circuit above correctly calculates the value of W. (*Hint*: Refer back to part (b) and use part (e).) Explain your answer. Do not simplify.

In the **Satisfiability** problem, we are given a proposition and asked whether there is *any* assignment of the variables that will make it true. In other words, a proposition is satisfiable iff its truth table has at least one row with value T.

Suppose we have access to software that solves the Satisfiability problem. Such a piece of software is referred to as a SAT solver. A SAT solver accepts a proposition A as input, and returns true iff A is satisfiable. Let's see how to use a SAT solver to finish checking our friend's work

(g) [1 Point] Explain why any proposition A is satisfiable iff A is not a contradiction.

(h) [1 Point] Explain how we can use a SAT solver to determine whether the circuit correctly calculates the value of W. Specifically, say what input we should pass to the SAT solver and how we can use its output to determine if the circuit is correct. (*Hint*: Use your answers to parts (f) and (g).)

8. Extra Credit: Go For the Gold (0 points)

Five pirates, called Ann, Brenda, Carla, Danielle and Emily, found a treasure of 100 gold coins. On their ship, they decide to split the coins using the following scheme:

- The first pirate in alphabetical order becomes the chief pirate.
- The chief proposes how to share the coins, and all other pirates (except the chief) vote for or against it.
- If 50% or more of the pirates vote for it, then the coins will be shared that way.
- Otherwise, the chief will be thrown overboard, and the process is repeated with the pirates that remain.

Thus, in the first round Ann is the chief: if her proposal is rejected, she is thrown overboard and Brenda becomes the chief, etc; if Ann, Brenda, Carla, and Danielle are thrown overboard, then Emily becomes the chief and keeps the entire treasure.

The pirates' first priority is to stay alive: they will act in such a way as to avoid death. If they can stay alive, they want to get as many coins as possible. Finally, they are a blood-thirsty bunch, if a pirate would get the same number of coins if she voted for or against a proposal, she will vote against so that the pirate who proposed the plan will be thrown overboard.

Assuming that all 5 pirates are intelligent (and aware that all the other pirates are just as aware, intelligent, and bloodthirsty), what will happen? Your solution should indicate which pirates die, and how many coins each of the remaining pirates receives.