CSE 390Z: Mathematics for Computation Workshop

Week 2 Workshop Solutions

Conceptual Review

(a) What are two different methods to show that two propositions are equivalent?

Solution:

Option 1: Write a truth table for each proposition and check that all rows have the same truth value.
Option 2: Use a chain of equivalences starting from one of the propositions and ending at the other.

(b) What is DNF form? What is CNF form? What about *canonical* DNF and *canonical* CNF?

Solution:

DNF and CNF are specific ways to write propositions.
DNF (Disjunctive Normal Form) is an "or of ands", and CNF (Conjunctive Normal Form) is an "and of ors".
Canonical DNF is an "or of minterms," one for each true row of the truth table. *Canonical* CNF is an "and of maxterms," one for each false row of the truth table. There are multiple (actually infinitely many) valid ways to write any proposition as a DNF or CNF, but only one valid *canonical* DNF and one valid *canonical* CNF expression.
We can also use DNFs and CNFs for boolean algebra expressions.

(c) When translating to predicate logic, how do you restrict to a smaller domain in a "for all"? How do you restrict to a smaller domain in a "there exists"?

Solution:

To restrict something quantified by a "for all", we use an **implication**.
To restrict something quantified by a "there exists", we use **and**.

For example, suppose the domain of discourse is all animals. We translate "all birds can fly" to $\forall x(\text{Bird}(x) \rightarrow \text{Fly}(x))$. We translate "there is a bird that can fly" to $\exists x(\text{Bird}(x) \wedge \text{Fly}(x))$.

(d) What is the difference between $\forall x \exists y(P(x, y))$ and $\exists y \forall x(P(x, y))$?

Solution:

$\forall x \exists y(P(x, y))$ means that every x has a y that satisfies $P(x, y)$. But, each x can have a different y. For example, if our domain of discourse is the integers, and $P(x, y)$ is $x + y = 0$, then $\forall x \exists y(P(x, y))$ means every integer x has a corresponding integer y such that $x + y = 0$. Since y can be different for every x, we can always pick $y = -x$, and the statement is true.

$\exists y \forall x(P(x, y))$ means that there is a special magical y that satisfies $P(x, y)$ for every x. Using our same $P(x, y)$ as above, the statement $\exists y \forall x(P(x, y))$ means there is a special integer y such that $x + y = 0$ for every integer x. This statement is of course false.

Another way to think about this is $\forall$ is a like a loop, and $\exists$ is like assigning a value to a single variable. Using the same $P(x, y)$ as above, but changing our domain to integers between -1000 and 1000,

for $\forall x \exists y$, you'd have something like this:

```
for (int x = -1000; x <= 1000; x++) {
    int y = -x;
    assert x + y == 0;
}
```

Since you assign a value to y inside the loop, the value can depend on the current value of x.

but for $\exists y \forall x$, you'd have something like this

```
final int y = 7; // 7 was random, the point is no number would work here.
for (int x = -1000; x <= 1000; x++) {
    assert x + y == 0;
}
```

You have to pick your value for y *before* you enter the loop, and you can't change it inside the loop. Since there is no value of y that would work, your code would crash and you would be sad.

1. DNFs and CNFs

Consider the following boolean functions $A(p,q,r)$ and $B(p,q,r)$.

p	q	r	$A(p,q,r)$	$B(p,q,r)$
1	1	1	0	1
1	1	0	0	1
1	0	1	1	1
1	0	0	0	0
0	1	1	1	0
0	1	0	1	1
0	0	1	0	1
0	0	0	0	0

Recall that to write the canonical DNF:

1. Read all the rows of the truth table where the boolean function evaluates to 1
2. Take the product of all the settings in a given 1 row
3. Take the sum of all of products from step 2

To write a canonical CNF:

1. Read all the rows of the truth table where the boolean function evaluates to 0
2. Take the sum of the negation of all of the settings in a given 0 row
3. Take the product of all of the sums from step 2

(a) Write the canonical DNF (sum of products) and canonical CNF (product of sums) expressions for $A(p,q,r)$.

Solution:

DNF: $pq'r + p'qr + p'qr'$

CNF: $(p'+q'+r')(p'+q'+r)(p'+q+r)(p+q+r')(p+q+r)$

(b) Write the canonical DNF (sum of products) and canonical CNF (product of sums) expressions for $B(p,q,r)$.

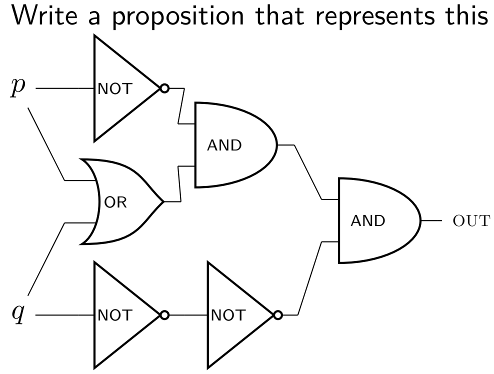
Solution:

DNF: $pqr + pqr' + pq'r + p'qr' + p'q'r$

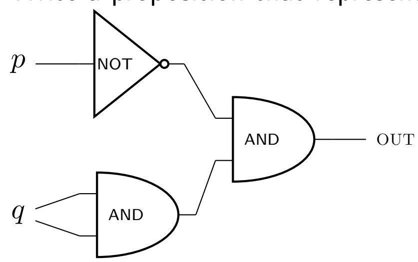
CNF: $(p'+q+r)(p+q'+r')(p+q+r)$

2. Circuits

(a) Write a proposition that represents this circuit:



(b) Write a proposition that represents this circuit:



Solution:

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

(b) $\neg p \wedge (q \wedge q)$

3. Equivalence Rules

We wish to prove that the following is true for all values of the propositions p and q (i.e. that it is a tautology):

$$\neg p \vee ((q \wedge p) \vee (\neg q \wedge p))$$

The following chain of equivalences does this, but it is missing citations for which rules are used. Fill in the blanks with names of the logic equivalences used at each step.

Hint: Reference the Logical Equivalences sheet under the Resources tab on the CSE 311 Course Website.

$$\begin{aligned}
\neg p \vee ((q \wedge p) \vee (\neg q \wedge p)) &\equiv \neg p \vee ((p \wedge q) \vee (\neg q \wedge p)) && \underline{\hspace{4cm}} \\
&\equiv \neg p \vee ((p \wedge q) \vee (p \wedge \neg q)) && \underline{\hspace{4cm}} \\
&\equiv \neg p \vee (p \wedge (q \vee \neg q)) && \underline{\hspace{4cm}} \\
&\equiv \neg p \vee (p \wedge T) && \underline{\hspace{4cm}} \\
&\equiv \neg p \vee p && \underline{\hspace{4cm}} \\
&\equiv p \vee \neg p && \underline{\hspace{4cm}} \\
&\equiv T && \underline{\hspace{4cm}}
\end{aligned}$$

Solution:

$$\begin{aligned}
\neg p \vee ((q \wedge p) \vee (\neg q \wedge p)) &\equiv \neg p \vee ((p \wedge q) \vee (\neg q \wedge p)) && \text{Commutativity} \\
&\equiv \neg p \vee ((p \wedge q) \vee (p \wedge \neg q)) && \text{Commutativity} \\
&\equiv \neg p \vee (p \wedge (q \vee \neg q)) && \text{Distributivity} \\
&\equiv \neg p \vee (p \wedge T) && \text{Negation} \\
&\equiv \neg p \vee p && \text{Identity} \\
&\equiv p \vee \neg p && \text{Commutativity} \\
&\equiv T && \text{Negation}
\end{aligned}$$

4. Equivalence: Propositional Logic

Prove $((p \wedge q) \rightarrow r) \equiv (p \rightarrow r) \vee (q \rightarrow r)$ via equivalences. Note that with propositional logic, you are expected to show all steps, including commutativity and associativity.

Solution:

$$
\begin{aligned}
(p \wedge q) \rightarrow r &\equiv \neg(p \wedge q) \vee r && \text{Law of Implication} \\
&\equiv (\neg p \vee \neg q) \vee r && \text{De Morgan's Law} \\
&\equiv (\neg p \vee \neg q) \vee (r \vee r) && \text{Idempotency} \\
&\equiv \neg p \vee (\neg q \vee (r \vee r)) && \text{Associativity} \\
&\equiv \neg p \vee ((\neg q \vee r) \vee r) && \text{Associativity} \\
&\equiv \neg p \vee (r \vee (\neg q \vee r)) && \text{Commutativity} \\
&\equiv (\neg p \vee r) \vee (\neg q \vee r) && \text{Associativity} \\
&\equiv (p \rightarrow r) \vee (\neg q \vee r) && \text{Law of Implication} \\
&\equiv (p \rightarrow r) \vee (q \rightarrow r) && \text{Law of Implication}
\end{aligned}
$$

5. Equivalence: Boolean Algebra

(a) Show $p' + (p \cdot q) + (q' \cdot p) = 1$ using boolean algebra axioms and theorems.

Solution:

$$
\begin{aligned}
p' + p \cdot q + q' \cdot p &= p' + p \cdot q + p \cdot q' && \text{Commutativity} \\
&= p' + p \cdot (q + q') && \text{Distributivity} \\
&= p' + p \cdot 1 && \text{Negation} \\
&= p' + p && \text{Identity} \\
&= p + p' && \text{Commutativity} \\
&= 1 && \text{Negation}
\end{aligned}
$$

(b) Show $(p' + q) \cdot (q + p) = q$ using boolean algebra axioms and theorems.

Solution:

$$
\begin{aligned}
(p' + q) \cdot (q + p) &= (q + p') \cdot (q + p) && \text{Commutativity} \\
&= q + (p' \cdot p) && \text{Distributivity} \\
&= q + (p \cdot p') && \text{Commutativity} \\
&= q + 0 && \text{Negation} \\
&= q && \text{Identity}
\end{aligned}
$$

6. Predicate Logic Warmup

Let the domain of discourse be all animals. Let Cat(x) ::= "x is a cat" and Blue(x) ::= "x is blue". Translate the following statements to English.

(a) $\forall x(\text{Cat}(x) \wedge \text{Blue}(x))$

Solution:

All animals are blue cats.

(b) $\forall x(\text{Cat}(x) \rightarrow \text{Blue}(x))$

Solution:

All cats are blue.

(c) $\exists x(\text{Cat}(x) \wedge \text{Blue}(x))$

Solution:

There is a blue cat.

Kabir translated the sentence "there exists a blue cat" to $\exists x(\text{Cat}(x) \rightarrow \text{Blue}(x))$. This is wrong! Let's understand why.

(d) Use the Law of Implications to rewrite Kabir's translation without the $\rightarrow$.

Solution:

$\exists x(\neg\text{Cat}(x) \vee \text{Blue}(x))$

(e) Translate the predicate from (d) back to English. How does this differ from the intended meaning?

Solution:

Translation: There is an animal that is not a cat, or is blue.

The difference: If there was even one non-cat animal in the universe (e.g. a single dog), this condition would be satisfied. Similarly, if there was even one blue animal in the universe, this condition would be satisfied. So, this is a very different condition than "there exists a blue cat".

7. English to Predicate Logic

Express the following sentences in predicate logic. The domain of discourse is penguins. You may use the following predicates: Love(x, y) ::= "x loves y", Dances(x) ::= "x dances", Sings(x) ::= "x sings" as well as $x = y$ and $x \neq y$

.

(a) There is a penguin that every penguin loves.

Solution:

$\exists x \forall y(\text{Loves}(y, x))$

(b) All penguins that sing love a penguin that does not sing.

Solution:

$\forall x(\text{Sings}(x) \to \exists y(\neg\text{Sings}(y) \land \text{Loves}(x, y)))$

(c) A penguin loves itself but hates (does not love) every other penguin.

Solution:

$\exists x(\text{Loves}(x, x) \land \forall y((y \neq x) \to \neg\text{Loves}(x, y)))$

(d) **Challenge:** There is exactly one penguin that dances.

Solution:

$\exists x(\text{Dances}(x) \land \forall y((y \neq x) \to \neg\text{Dances}(y)))$

or, equivalently:

$\exists x(\text{Dances}(x) \land \forall y(\text{Dances}(y) \to (x = y)))$

or, equivalently:

$\exists x \forall y[(y = x) \leftrightarrow \text{Dances}(y)]$

8. Predicate Logic to English

Translate the following sentences to English. Assume the same predicates and domain of discourse as the previous problem.

(a) $\neg\exists x(\text{Dances}(x))$

Solution:

No penguins dance.

(b) $\exists x \forall y(\text{Loves}(x, y))$

Solution:

A penguin loves all penguins.

(c) $\forall x[\text{Dances}(x) \to \exists y(\text{Loves}(y, x))]$

Solution:

Penguins that dance have a penguin that loves them.

(d) $\exists x \forall y[(\text{Dances}(y) \land \text{Sings}(y)) \to \text{Loves}(x, y)]$

Solution:

A penguin loves all penguins that dance and sing.