CSE 311: Foundations of Computing I

Section 2: Logical Equivalences, Predicate Translation Solutions

1. Boolean Algebra and Digital Circuits

Consider the following propositional logic expression:

$$\neg(\neg p \lor (p \land \neg r))$$

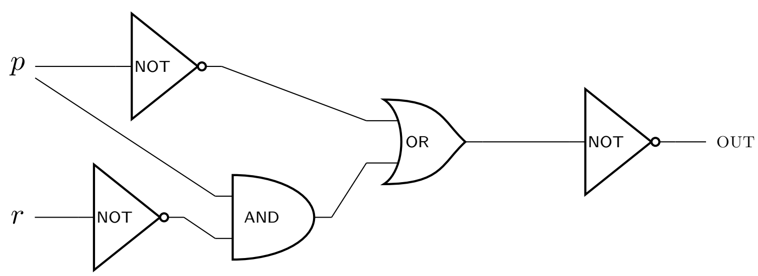
(a) Translate the proposition to Boolean Algebra notation. Do not simplify the expression.

Solution:

$$(p' + (p \cdot r'))'$$

(b) Write the proposition as a Digital Circuit.

Solution:



2. Logical Equivalences

Prove that each of the following pairs of propositional formulas are equivalent using logical equivalences.

(a) $p \to q \equiv \neg(p \land \neg q)$

Solution:

$$\begin{aligned} p \to q &\equiv \neg p \lor q && \text{Law of Implication} \\ &\equiv \neg p \lor \neg\neg q && \text{Double Negation} \\ &\equiv \neg(p \land \neg q) && \text{DeMorgan's Law} \end{aligned}$$

(b) $\neg p \to (s \to r) \equiv s \to (p \lor r)$

Solution:

$$\begin{aligned}
\neg p \to (s \to r) &\equiv \neg\neg p \vee (s \to r) && \text{Law of Implication} \\
&\equiv p \vee (s \to r) && \text{Double Negation} \\
&\equiv p \vee (\neg s \vee r) && \text{Law of Implication} \\
&\equiv (p \vee \neg s) \vee r && \text{Associativity} \\
&\equiv (\neg s \vee p) \vee r && \text{Commutativity} \\
&\equiv \neg s \vee (p \vee r) && \text{Associativity} \\
&\equiv s \to (p \vee r) && \text{Law of Implication}
\end{aligned}$$

(c) $\neg p \vee ((q \wedge p) \vee (\neg q \wedge p)) \equiv \mathsf{T}$

Solution:

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

(d) $((p \wedge q) \to r) \equiv (p \to r) \vee (q \to r)$

Solution:

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

3. DNFs and CNFs

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

p	q	r	$A(p, q, r)$	$B(p, q, r)$
T	T	T	F	T
T	T	F	F	T
T	F	T	T	T
T	F	F	F	F
F	T	T	T	F
F	T	F	T	T
F	F	T	F	T
F	F	F	F	F

(a) Write the DNF (ORs of ANDs) and CNF (ANDs of ORs) expressions for $A(p, q, r)$.

Solution:

DNF: $(p \wedge \neg q \wedge r) \vee (\neg p \wedge q \wedge r) \vee (\neg p \wedge q \wedge \neg r)$

CNF: $(\neg p \vee \neg q \vee \neg r) \wedge (\neg p \vee \neg q \vee r) \wedge (\neg p \vee q \vee r) \wedge (p \vee q \vee \neg r) \wedge (p \vee q \vee r)$

(b) Write the DNF (ORs of ANDs) and CNF (ANDs of ORs) expressions for $B(p, q, r)$.

Solution:

DNF: $(p \wedge q \wedge r) \vee (p \wedge q \wedge \neg r) \vee (p \wedge \neg q \wedge r) \vee (\neg p \wedge q \wedge \neg r) \vee (\neg p \wedge \neg q \wedge r)$

CNF: $(\neg p \vee q \vee r) \wedge (p \vee \neg q \vee \neg r) \wedge (p \vee q \vee r)$

4. Predicate Logic to English

Translate the following sentences to English. The domain of discourse is penguins. The predicates are defined as follows: Love(x, y) ::= "x loves y", Dances(x) ::= "x dances", Sings(x) ::= "x sings".

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

Solution:

No penguins dance.

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

Solution:

There is a penguin that loves all penguins.

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

Solution:

All penguins that dance have a penguin that loves them.

(d) $\exists x \forall y((\text{Dances}(y) \wedge \text{Sings}(y)) \rightarrow \text{Loves}(x, y))$

Solution:

There exists a penguin that loves all penguins who dance and sing.

5. English to Predicate Logic

Translate each of the following sentences into logical notation. These translations require some of our quantifier tricks. You may use the operators $+$ and $\cdot$ which take two numbers as input and evaluate to their sum or product, respectively. Remember:

- To restrict the domain under a $\forall$ quantifier, add a hypothesis to an implication.
- To restrict the domain under an $\exists$ quantifier, AND in the restriction.
- If you want variables to be different, you have to explicitly require them to be not equal.

(a) Domain: Positive integers; Predicates: Even, Prime, Equal
"There is a positive integer that is prime and even."

Solution:

$\exists x(\text{Prime}(x) \wedge \text{Even}(x))$

(b) Domain: Positive integers; Predicates: Even, Prime, Equal
"If a positive integer is even, its square is also even."

Solution:

$\forall x(\text{Even}(x) \rightarrow \text{Even}(x^2))$

(c) Domain: Real numbers; Predicates: Even, Prime, Equal
"There are two prime numbers that sum to an even number."

Solution:

$\exists x \exists y(\text{Prime}(x) \wedge \text{Prime}(y) \wedge \text{Even}(x+y))$

(d) Domain: Positive integers; Predicates: Even, Prime, Equal
"There is only one positive integer that is prime and even."

Solution:

$\exists x(\text{Prime}(x) \wedge \text{Even}(x) \wedge \forall y[\neg\text{Equal}(x,y) \rightarrow \neg(\text{Even}(y) \wedge \text{Prime}(y))])$
OR equivalently
$\exists x(\text{Prime}(x) \wedge \text{Even}(x) \wedge \forall y[(\text{Even}(y) \wedge \text{Prime}(y)) \rightarrow \text{Equal}(x,y)])$

(e) Domain: Real numbers; Predicates: Even, Prime, Equal
"The product of two distinct prime numbers is not prime."

Solution:

$\forall x \forall y([\text{Prime}(x) \wedge \text{Prime}(y) \wedge \neg\text{Equal}(x,y)] \rightarrow \neg\text{Prime}(xy))$

(f) Domain: Real numbers; Predicates: Even, Prime, Equal, Positive, Greater, Integer
"For every positive integer, there is a greater even integer"

Solution:

$\forall x(\text{Positive}(x) \land \text{Integer}(x) \rightarrow [\exists y(\text{Integer}(y) \land \text{Even}(y) \land \text{Greater}(y, x))])$
OR equivalently
$\forall x \exists y(\text{Positive}(x) \land \text{Integer}(x) \rightarrow (\text{Integer}(y) \land \text{Even}(y) \land \text{Greater}(y, x)))$

6. Quantifier Switch

Consider the following pairs of sentences. For each pair, determine if one implies the other, if they are equivalent, or neither.

(a) $\forall x\ \forall y\ P(x, y)$ $\qquad$ $\forall y\ \forall x\ P(x, y)$

Solution:

These sentences are the same; switching universal quantifiers makes no difference.

(b) $\exists x\ \exists y\ P(x, y)$ $\qquad$ $\exists y\ \exists x\ P(x, y)$

Solution:

These sentences are the same; switching existential quantifiers makes no difference.

(c) $\forall x\ \exists y\ P(x, y)$ $\qquad$ $\forall y\ \exists x\ P(x, y)$

Solution:

These are only the same if P is symmetric (i.e., the order of the arguments doesn't matter). If the order of the arguments does matter, then these are different statements. For instance, if $P(x, y)$ is "$x < y$", then the first statement says "for every x, there is a corresponding y such that $x < y$", whereas the second says "for every y, there is a corresponding x such that $x < y$". In other words, in the first statement y is a function of x, and in the second x is a function of y.

If your domain of discourse is "positive integers", for example, the first is true and the second is false; but for "negative integers" the second is true while the first is false.

(d) $\forall x\ \exists y\ P(x, y)$ $\qquad$ $\exists x\ \forall y\ P(x, y)$

Solution:

These two statements are usually different.

(e) $\forall x\ \exists y\ P(x, y)$ $\qquad$ $\exists y\ \forall x\ P(x, y)$

Solution:

The second statement is "stronger" than the first (that is, the second implies the first). For the first, y is allowed to depend on x. For the second, one specific y must work for all x. Thus if the second is true, whatever value of y makes it true, can also be plugged in for y in the first statement for every x. On the other hand, if the first statement is true, it might be that different y's work for the different x's and no single value of y exists to make the latter true.

As an example, let your domain of discourse be positive real numbers, and let $P(x, y)$ be $xy = 1$. The first statement is true (always take y to be $1/x$, which is another positive real number). The second statement is not true; it asks for a single number that always makes the product 1.