

CSE 311: Foundations of Computing I

Section 2: Equivalences and Predicate Logic Solutions

1. Equivalences

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

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

Solution:

$p \leftrightarrow q$	\equiv	$(p \rightarrow q) \wedge (q \rightarrow p)$	[iff is two implications]
	\equiv	$(\neg p \vee q) \wedge (q \rightarrow p)$	[Law of Implication]
	\equiv	$(\neg p \vee q) \wedge (\neg q \vee p)$	[Law of Implication]
	\equiv	$((\neg p \vee q) \wedge \neg q) \vee ((\neg p \vee q) \wedge p)$	[Distributivity]
	\equiv	$(\neg q \wedge (\neg p \vee q)) \vee ((\neg p \vee q) \wedge p)$	[Commutativity]
	\equiv	$((\neg q \wedge \neg p) \vee (\neg q \wedge q)) \vee ((\neg p \vee q) \wedge p)$	[Distributivity]
	\equiv	$((\neg p \wedge \neg q) \vee (\neg q \wedge q)) \vee ((\neg p \vee q) \wedge p)$	[Commutativity]
	\equiv	$((\neg p \wedge \neg q) \vee (q \wedge \neg q)) \vee ((\neg p \vee q) \wedge p)$	[Commutativity]
	\equiv	$((\neg p \wedge \neg q) \vee (q \wedge \neg q)) \vee (p \wedge (\neg p \vee q))$	[Commutativity]
	\equiv	$((\neg p \wedge \neg q) \vee (q \wedge \neg q)) \vee ((p \wedge \neg p) \vee (p \wedge q))$	[Distributivity]
	\equiv	$((\neg p \wedge \neg q) \vee F) \vee ((p \wedge \neg p) \vee (p \wedge q))$	[Negation]
	\equiv	$((\neg p \wedge \neg q) \vee F) \vee (F \vee (p \wedge q))$	[Negation]
	\equiv	$(\neg p \wedge \neg q) \vee (F \vee (p \wedge q))$	[Identity]
	\equiv	$(\neg p \wedge \neg q) \vee ((p \wedge q) \vee F)$	[Commutativity]
	\equiv	$(\neg p \wedge \neg q) \vee (p \wedge q)$	[Identity]
	\equiv	$(p \wedge q) \vee (\neg p \wedge \neg q)$	[Commutativity]

(b) $\neg p \rightarrow (q \rightarrow r)$ $q \rightarrow (p \vee r)$

Solution:

$\neg p \rightarrow (q \rightarrow r)$	\equiv	$\neg \neg p \vee (q \rightarrow r)$	[Law of Implication]
	\equiv	$p \vee (q \rightarrow r)$	[Double Negation]
	\equiv	$p \vee (\neg q \vee r)$	[Law of Implication]
	\equiv	$(p \vee \neg q) \vee r$	[Associativity]
	\equiv	$(\neg q \vee p) \vee r$	[Commutativity]
	\equiv	$\neg q \vee (p \vee r)$	[Associativity]
	\equiv	$q \rightarrow (p \vee r)$	[Law of Implication]

2. Non-equivalence

Prove that the following pairs of propositional formulae are not equivalent by finding inputs they differ on.

(a) $p \rightarrow q$ $q \rightarrow p$

Solution:

When $p = T$ and $q = F$, then $p \rightarrow q \equiv F$, but $q \rightarrow p \equiv T$.

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

Solution:

When $p = F$ and $r = F$, then $p \rightarrow (q \wedge r) \equiv T$, but $(p \rightarrow q) \wedge r \equiv F$.

3. Boolean Algebra

For each of the following parts, write the logical expression using boolean algebra operators. Then, simplify it using axioms and theorems of boolean algebra.

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

Solution:

First, we replace \neg, \vee , and \wedge . This gives us $p' + q' + pq$; note that the parentheses are not necessary in boolean algebra, because the operations are all commutative and associative. We can use DeMorgan's laws to get the slightly simpler $(pq)' + pq$. Then, we can use commutativity to get $pq + (pq)'$ and complementarity to get 1. (Note that this is another way of saying the formula is a tautology.)

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

Solution:

First, we replace \neg, \vee , and \wedge with their corresponding boolean operators, giving us $(p + (qp))'$. Applying DeMorgan's laws once gives us $p'(qp)'$, and a second time gives us $p'(q' + p')$, which is $p'(p' + q')$ by commutativity. By absorption, this is simply p' .

4. Canonical Forms

Consider the boolean functions $F(A, B, C)$ and $G(A, B, C)$ specified by the following truth table:

A	B	C	$F(A, B, C)$	$G(A, B, C)$
1	1	1	1	0
1	1	0	1	1
1	0	1	0	0
1	0	0	0	0
0	1	1	1	1
0	1	0	1	0
0	0	1	0	1
0	0	0	1	0

(a) Write the DNF and CNF expressions for $F(A, B, C)$.

Solution:**DNF:** $ABC + ABC' + A'BC + A'BC' + A'B'C'$ **CNF:** $(A' + B + C')(A' + B + C)(A + B + C')$

- (b) Write the DNF and CNF expressions for
- $G(A, B, C)$
- .

Solution:**DNF:** $ABC' + A'BC + A'B'C$ **CNF:** $(A' + B' + C')(A' + B + C')(A' + B + C)(A + B' + C)(A + B + C)$

5. Translate to Logic

Express each of these system specifications using predicate, quantifiers, and logical connectives.

- (a) Every user has access to an electronic mailbox.

Solution:Let the domain be users and mailboxes. Let $User(x)$ be “ x is a user”, let $Mailbox(y)$ be “ y is a mailbox”, and let $Access(x, y)$ be “ x has access to y ”.

$$\forall x (User(x) \rightarrow (\exists y (Mailbox(y) \wedge Access(x, y))))$$

- (b) The system mailbox can be accessed by everyone in the group if the file system is locked.

Solution:Let the domain be people in the group. Let $Access(x, y)$ be “ x has access to y ”. Let $FileSystemLocked$ be the proposition “the file system is locked.” Let $SystemMailbox$ be the constant that is the system mailbox.

$$FileSystemLocked \rightarrow \forall x Access(x, SystemMailbox)$$

- (c) The firewall is in a diagnostic state only if the proxy server is in a diagnostic state.

Solution:Let the domain be all applications. Let $Firewall(x)$ be “ x is the firewall”, and let $ProxyServer(x)$ be “ x is the proxy server.” Let $Diagnostic(x)$ be “ x is in a diagnostic state”.

$$\forall x \forall y ((Firewall(x) \wedge Diagnostic(x)) \rightarrow (ProxyServer(y) \rightarrow Diagnostic(y)))$$

- (d) At least one router is functioning normally if the throughput is between 100kbps and 500 kbps and the proxy server is not in diagnostic mode.

Solution:

Let the domain be all applications and routers. Let $\text{Router}(x)$ be “ x is a router”, and let $\text{ProxyServer}(x)$ be “ x is the proxy server.” Let $\text{Diagnostic}(x)$ be “ x is in a diagnostic state”. Let ThroughputNormal be “the throughput is between 100kbps and 500 kbps”. Let $\text{Functioning}(y)$ be “ y is functioning normally”.

$$(\text{ThroughputNormal} \wedge \forall x (\neg \text{ProxyServer}(x) \vee \neg \text{Diagnostic}(x))) \rightarrow \exists y (\text{Router}(y) \wedge \text{Functioning}(y))$$

6. Translate to English

Translate these system specifications into English where $F(p)$ is “Printer p is out of service”, $B(p)$ is “Printer p is busy”, $L(j)$ is “Print job j is lost,” and $Q(j)$ is “Print job j is queued”. Let the domain be all printers.

(a) $\exists p (F(p) \wedge B(p)) \rightarrow \exists j L(j)$

Solution:

If at least one printer is busy and out of service, then at least one job is lost.

(b) $(\forall p B(p)) \rightarrow (\exists j Q(j))$

Solution:

If all printers are busy, then there is a queued job.

(c) $\exists j (Q(j) \wedge L(j)) \rightarrow \exists p F(p)$

Solution:

If there is a queued job that is lost, then a printer is out of service.

(d) $(\forall p B(p) \wedge \forall j Q(j)) \rightarrow \exists j L(j)$

Solution:

If all printers are busy and all jobs are queued, then there is some lost job.