Section 02: Solutions

1. Equivalences

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

(a) $p \leftrightarrow r \equiv (p \wedge r) \vee (\neg p \wedge \neg r)$ You may use the rule $p \leftrightarrow r \equiv (p \rightarrow r) \wedge (r \rightarrow p)$. **Solution:**

$$
\begin{array}{lcll}
p \leftrightarrow r & \equiv & (p \rightarrow r) \wedge (r \rightarrow p) & \text{[iff is two implications]} \\
 & \equiv & (\neg p \vee r) \wedge (r \rightarrow p) & \text{[Law of Implication]} \\
 & \equiv & (\neg p \vee r) \wedge (\neg r \vee p) & \text{[Law of Implication]} \\
 & \equiv & ((\neg p \vee r) \wedge \neg r) \vee ((\neg p \vee r) \wedge p) & \text{[Distributivity]} \\
 & \equiv & (\neg r \wedge (\neg p \vee r)) \vee ((\neg p \vee r) \wedge p) & \text{[Commutativity]} \\
 & \equiv & ((\neg r \wedge \neg p) \vee (\neg r \wedge r)) \vee ((\neg p \vee r) \wedge p) & \text{[Distributivity]} \\
 & \equiv & ((\neg p \wedge \neg r) \vee (\neg r \wedge r)) \vee ((\neg p \vee r) \wedge p) & \text{[Commutativity]} \\
 & \equiv & ((\neg p \wedge \neg r) \vee (r \wedge \neg r)) \vee ((\neg p \vee r) \wedge p) & \text{[Commutativity]} \\
 & \equiv & ((\neg p \wedge \neg r) \vee (r \wedge \neg r)) \vee (p \wedge (\neg p \vee r)) & \text{[Commutativity]} \\
 & \equiv & ((\neg p \wedge \neg r) \vee (r \wedge \neg r)) \vee ((p \wedge \neg p) \vee (p \wedge r)) & \text{[Distributivity]} \\
 & \equiv & ((\neg p \wedge \neg r) \vee F) \vee ((p \wedge \neg p) \vee (p \wedge r)) & \text{[Negation]} \\
 & \equiv & ((\neg p \wedge \neg r) \vee F) \vee (F \vee (p \wedge r)) & \text{[Negation]} \\
 & \equiv & (\neg p \wedge \neg r) \vee (F \vee (p \wedge r)) & \text{[Identity]} \\
 & \equiv & (\neg p \wedge \neg r) \vee ((p \wedge r) \vee F) & \text{[Commutativity]} \\
 & \equiv & (\neg p \wedge \neg r) \vee (p \wedge r) & \text{[Identity]} \\
 & \equiv & (p \wedge r) \vee (\neg p \wedge \neg r) & \text{[Commutativity]}
\end{array}
$$

The “intuition” for this problem is hard to see. After the first application of the distributive law – when we have $((\neg p \vee r) \wedge \neg r) \vee ((\neg p \vee r) \wedge p)$ a case analysis would indicate that $\vee r$ isn’t helping on the left subexpression (if r is true, then the expression is false, if r is false, then it doesn’t help in $\neg p \vee r$). By distributing, we are able to simplify that term out.

Similarly, a case analysis on the right subexpression indicates $\vee r$ is not contributing to the expression (and distributing lets us see and simplify).

(b) $\neg s \rightarrow (q \rightarrow r) \equiv q \rightarrow (s \vee r)$ **Solution:**

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

2. Canonical Forms

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

A	B	C	$F(A,B,C)$	$G(A,B,C)$
T	T	T	T	F
T	T	F	T	T
T	F	T	F	F
T	F	F	F	F
F	T	T	T	T
F	T	F	T	F
F	F	T	F	T
F	F	F	T	F

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

DNF: $(A \wedge B \wedge C) \vee (A \wedge B \wedge \neg C) \vee (\neg A \wedge B \wedge C) \vee (\neg A \wedge B \wedge \neg C) \vee (\neg A \wedge \neg B \wedge \neg C)$
CNF: $(\neg A \vee B \vee \neg C) \wedge (\neg A \vee B \vee C) \wedge (A \vee B \vee \neg C)$

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

DNF: $(A \wedge B \wedge \neg C) \vee (\neg A \wedge B \wedge C) \vee (\neg A \wedge \neg B \wedge C)$
CNF: $(\neg A \vee \neg B \vee \neg C) \wedge (\neg A \vee B \vee \neg C) \wedge (\neg A \vee B \vee C) \wedge (A \vee \neg B \vee C) \wedge (A \vee B \vee C)$

3. Translate to Logic

Express each of these system specifications using predicates, quantifiers, and logical connectives. For some of these problems, more than one translation will be reasonable depending on your choice of predicates.

(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".

$$\forall x\ (\texttt{User}(x) \to (\exists y\ (\texttt{Mailbox}(y) \land \texttt{Access}(x,y))))$$

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

Solution 1: Let the domain be people in the group. Let $\texttt{CanAccessSM}(x)$ be "x has access to the system mailbox". Let p be the proposition "the file system is locked."

$$p \to \forall x\ \texttt{CanAccessSM}(x).$$

Solution2: Let the domain be people and mailboxes and use $\texttt{Access}(x,y)$ as defined in the solution to part (a), and then also add $\texttt{InGroup}(x)$ for "x is in the group", and let SystemMailBox be the name for the system mailbox. Then the translation becomes

$$\texttt{FileSystemLocked} \to \forall x\ (\texttt{InGroup}(x) \to \texttt{Access}(x, \texttt{SystemMailBox})).$$

(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 $\texttt{Firewall}(x)$ be "x is the firewall", and let $\texttt{ProxyServer}(x)$ be "x is the proxy server." Let $\texttt{Diagnostic}(x)$ be "x is in a diagnostic state".

$$\forall x\ \forall y\ ((\texttt{Firewall}(x) \land \texttt{Diagnostic}(x)) \to (\texttt{ProxyServer}(y) \to \texttt{Diagnostic}(y))$$

(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 $\texttt{Router}(x)$ be "x is a router", and let $\texttt{ProxyServer}(x)$ be "x is the proxy server." Let $\texttt{Diagnostic}(x)$ be "x is in a diagnostic state". Let p be "the throughput is between 100kbps and 500 kbps". Let $\texttt{Functioning}(y)$ be "y is functioning normally".

$$(p \land \forall x\ (\neg\texttt{ProxyServer}(x) \lor \neg\texttt{Diagnostic}(x))) \to \exists y\ (\texttt{Router}(y) \land \texttt{Functioning}(y))$$

4. 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 and all print jobs.

(a) $\exists p\ (F(p) \land B(p)) \to \exists j\ L(j)$ **Solution:**

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

(b) $(\forall j\ B(j)) \to (\exists p\ Q(p))$ **Solution:**

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

(c) $\exists j\ (Q(j) \wedge L(j)) \to \exists p\ F(p)$ **Solution:**

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

(d) $(\forall p\ B(p) \wedge \forall j\ Q(j)) \to \exists j\ L(j)$ **Solution:**

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

5. ctrl-z

Translate these logical expressions to English. For each of the translations, assume that domain restriction is being used and take that into account in your English versions.

Let your domain be all UW Students. Predicates $\texttt{143Student}(x)$ and $\texttt{311Student}(x)$ mean the student is in CSE 143 and 311, respectively. $\texttt{BioMajor}(x)$ means x is a bio major, `DidHomeworkOne(x)` means the student did homwork 1 (of 311). Finally $\texttt{KnowsJava}(x)$ and $\texttt{KnowsDeMorgan}(x)$ mean x knows Java and knows DeMorgan's Laws, respectively.

(a) $\forall x(\texttt{143Student}(x) \to \texttt{KnowsJava}(x))$ **Solution:**

> Every 143 student knows java.
>
> ”If a UW student is a 143 student, then they know java” is a valid translation of the original sentence, but it is not taking advantage of the domain restriction.

(b) $\exists x(\texttt{143Student}(x) \wedge \texttt{BioMajor}(x))$ **Solution:**

> “There is a 143 student who is a bio major”
>
> “There is a UW student who is a 143 student and is a bio major” is a valid translation of the original sentence, but is not taking advantage of the domain restriction.

(c) $\forall x([\texttt{311Student}(x) \wedge \texttt{DidHomeworkOne(x)}] \to \texttt{KnowsDeMorgan}(x))$ **Solution:**

> If a 311 student does homework one then they know DeMorgan's Laws.

6. Domain Restriction Negated

When we negate a sentence with a domain restriction, the restriction itself remains intact (i.e. not negated) after we have fully simplified. That should make sense; if I claim something is true for every even number, I won't be convinced by you showing you an odd number. In this problem, you'll do the algebra to see why.

(a) Consider the statement $\neg\exists x\exists y(\texttt{Domain1}(x) \wedge \texttt{Domain2}(y) \wedge [P(x,y) \wedge Q(x,y)])$. We know that we should end up with $\forall x\forall y(\texttt{Domain1}(x) \wedge \texttt{Domain2}(y) \to [\neg P(x,y) \vee \neg Q(x,y)])$ (that is flip the quantifiers, rewrite the domain restriction, and negate the other requirements).

But it can help to see the full algebra written out – write out a step-by-step simplification to get the simplified form (you don't have to label with rules). **Solution:**

$$\begin{aligned}
\neg\exists x\exists y(\texttt{Domain1}(x) \wedge \texttt{Domain2}(y) \wedge [P(x,y) \wedge Q(x,y)]) &\equiv \forall x\neg[\exists y(\texttt{Domain1}(x) \wedge \texttt{Domain2}(y) \wedge [P(x,y) \wedge Q(x,y)])] \\
&\equiv \forall x\forall y\neg[(\texttt{Domain1}(x) \wedge \texttt{Domain2}(y) \wedge [P(x,y) \wedge Q(x,y)])] \\
&\equiv \forall x\forall y(\neg\texttt{Domain1}(x) \vee \neg\texttt{Domain2}(y) \vee \neg[P(x,y) \wedge Q(x,y)]) \\
&\equiv \forall x\forall y(\neg\texttt{Domain1}(x) \vee \neg\texttt{Domain2}(y) \vee [\neg P(x,y) \vee \neg Q(x,y)]) \\
&\equiv \forall x\forall y([\neg\texttt{Domain1}(x) \vee \neg\texttt{Domain2}(y)] \vee [\neg P(x,y) \vee \neg Q(x,y)]) \\
&\equiv \forall x\forall y(\neg[\texttt{Domain1}(x) \wedge \texttt{Domain2}(y)] \vee [\neg P(x,y) \vee \neg Q(x,y)]) \\
&\equiv \forall x\forall y([\texttt{Domain1}(x) \wedge \texttt{Domain2}(y)] \to [\neg P(x,y) \vee \neg Q(x,y)])
\end{aligned}$$

(b) Now do the same process for: $\neg\forall x\forall y(\texttt{Domain1}(x) \wedge \texttt{Domain2}(y) \to [P(x,y) \wedge Q(x,y)])$ **Solution:**

$$\begin{aligned}
&\neg\forall x\forall y(\texttt{Domain1}(x) \wedge \texttt{Domain2}(y) \to [P(x,y) \wedge Q(x,y)]) \\
\equiv&\exists x\neg[\forall y(\texttt{Domain1}(x) \wedge \texttt{Domain2}(y) \to [P(x,y) \wedge Q(x,y)])] \\
\equiv&\exists x\exists y\neg[(\texttt{Domain1}(x) \wedge \texttt{Domain2}(y) \to [P(x,y) \wedge Q(x,y)])] \\
\equiv&\exists x\exists y\neg[\neg(\texttt{Domain1}(x) \wedge \texttt{Domain2}(y)) \vee [P(x,y) \wedge Q(x,y)])] \\
\equiv&\exists x\exists y\neg[(\neg\texttt{Domain1}(x) \vee \neg\texttt{Domain2}(y)) \vee [P(x,y) \wedge Q(x,y)])] \\
\equiv&\exists x\exists y[(\neg\neg\texttt{Domain1}(x) \wedge \neg\neg\texttt{Domain2}(y)) \wedge \neg[P(x,y) \wedge Q(x,y)])] \\
\equiv&\exists x\exists y[(\texttt{Domain1}(x) \wedge \texttt{Domain2}(y)) \wedge \neg[P(x,y) \wedge Q(x,y)])] \\
\equiv&\exists x\exists y[\texttt{Domain1}(x) \wedge \texttt{Domain2}(y) \wedge [\neg P(x,y) \vee \neg Q(x,y)])]
\end{aligned}$$