CS536 Science of Programming
Fall 2024
Assignment 1 Sample Solution Sketches

1. (a) In general, $e_1 = e_2$ does not logically imply $e_1 \equiv e_2$. A counterexample can be $2+2=4$ but $2+2 \not\equiv 4$.

   (b) In general, $e_1 \neq e_2$ logically implies $e_1 \not\equiv e_2$. We know that syntactic equality implies semantic equality, or in other words, $e_1 \equiv e_2 \Rightarrow e_1 = e_2$, the contrapositive of this statement is $e_1 \neq e_2 \Rightarrow e_1 \not\equiv e_2$, which is also true.

2. (a) In the following truth table, we can see that $(p \vee q) \wedge q$ and q always have the same truth value, so they are logically equivalent to each other.

p	q	$p \vee q$	$(p \vee q) \wedge q$
T	T	T	T
T	F	T	F
F	T	T	T
F	F	F	F

   (b) In the following truth table, we can see that $\neg(p \leftrightarrow q)$ and $\neg p \leftrightarrow q$ always have the same truth value, so they are logically equivalent to each other.

p	q	$p \leftrightarrow q$	$\neg(p \leftrightarrow q)$	$\neg p$	$\neg p \leftrightarrow q$
T	T	T	F	F	F
T	F	F	T	F	T
F	T	F	T	T	T
F	F	T	F	T	F

   (c) In the following truth table, we can see the column that represents the truth value of $\neg p \wedge (p \vee q) \rightarrow q$ is always true, so $\neg p \wedge (p \vee q) \rightarrow q$ is logically equivalent to T.

p	q	$\neg p$	$p \vee q$	$\neg p \wedge (p \vee q)$	$\neg p \wedge (p \vee q) \rightarrow q$
T	T	F	T	F	T
T	F	F	T	F	T
F	T	T	T	T	T
F	F	T	F	F	T

3. (a) Here I start from the left-hand side of this equality.

$$
\begin{aligned}
(p \rightarrow q) \vee (p \rightarrow r) \quad & \Leftrightarrow (\neg p \vee q) \vee (p \rightarrow r) && \text{Definition of } \rightarrow \\
& \Leftrightarrow (\neg p \vee q) \vee (\neg p \vee r) && \text{Definition of } \rightarrow \\
& \Leftrightarrow (\neg p \vee \neg p) \vee (q \vee r) && \text{Commutativity and Associativity of } \vee \\
& \Leftrightarrow \neg p \vee (q \vee r) && \text{Idempotency} \\
& \Leftrightarrow p \rightarrow (q \vee r) && \text{Definition of } \rightarrow
\end{aligned}
$$

(b) Here I start from the left-hand side of this equality.

$$\begin{aligned}
(p \vee q) \wedge \neg q \quad &\Leftrightarrow (p \wedge \neg q) \vee (q \wedge \neg q) && \text{Distributivity of } \wedge \\
&\Leftrightarrow (p \wedge \neg q) \vee F && \text{Contradiction} \\
&\Leftrightarrow p \wedge \neg q && \text{Identity} \\
&\Leftrightarrow \neg(p \rightarrow q) && \text{Negation of Implication}
\end{aligned}$$

(c) Here I start from the left-hand side of this equality.

$$\begin{aligned}
(p \rightarrow q) \wedge (\neg p \rightarrow q) \quad &\Leftrightarrow (\neg p \vee q) \wedge (\neg p \rightarrow q) && \text{Definition of } \rightarrow \\
&\Leftrightarrow (\neg p \vee q) \wedge (p \vee q) && \text{Definition of } \rightarrow \text{ and Double Negation} \\
&\Leftrightarrow (\neg p \wedge p) \vee q && \text{Distributivity of } \vee \\
&\Leftrightarrow q && \text{Contradiction and Identity}
\end{aligned}$$

4. (a)

$$\begin{aligned}
\neg(p \wedge q) \wedge p \quad &\Leftrightarrow (\neg p \vee \neg q) \wedge p && \text{DeMorgan's Law} \\
&\Leftrightarrow (\neg p \wedge p) \vee (\neg q \wedge p) && \text{Distributivity of } \wedge \\
&\Leftrightarrow \neg q \wedge p && \text{Contradiction and Identity} \\
&\Rightarrow \neg q && \text{and-elimination}
\end{aligned}$$

(b)

$$\begin{aligned}
p \wedge q \vee q \wedge r \quad &\Rightarrow p \vee q \wedge r && \text{and-elimination} \\
&\Rightarrow p \vee q && \text{and-elimination} \\
&\Rightarrow p \vee q \vee r && \text{or-introduction}
\end{aligned}$$

(c)

$$\begin{aligned}
(p \rightarrow q) \wedge (\neg p \rightarrow r) \quad &\Leftrightarrow T \wedge (p \rightarrow q) \wedge (\neg p \rightarrow r) && \text{Identity} \\
&\Leftrightarrow (p \vee \neg p) \wedge (p \rightarrow q) \wedge (\neg p \rightarrow r) && \text{Excluded Middle} \\
&\Leftrightarrow p \wedge (p \rightarrow q) \wedge (\neg p \rightarrow r) \vee \\
&\qquad \neg p \wedge (p \rightarrow q) \wedge (\neg p \rightarrow r) && \text{Distributivity of } \wedge \\
&\Rightarrow q \wedge (\neg p \rightarrow r) \vee \\
&\qquad \neg p \wedge (p \rightarrow q) \wedge (\neg p \rightarrow r) && \text{Modus Ponens} \\
&\Rightarrow q \wedge (\neg p \rightarrow r) \vee \\
&\qquad r \wedge (p \rightarrow q) && \text{Modus Ponens} \\
&\Rightarrow q \vee r \wedge (p \rightarrow q) && \text{and-elimination} \\
&\Rightarrow q \vee r && \text{and-elimination}
\end{aligned}$$

5. Here are the states that satisfy $p \leftrightarrow q \leftrightarrow r$:

(a) $\sigma = \{p = T, q = T, r = T\}$, since $\sigma(q \leftrightarrow r) = T$ and $\sigma(p) \leftrightarrow T = T$.
(b) $\sigma = \{p = T, q = F, r = F\}$, since $\sigma(q \leftrightarrow r) = T$ and $\sigma(p) \leftrightarrow T = T$.
(c) $\sigma = \{p = F, q = F, r = T\}$, since $\sigma(q \leftrightarrow r) = F$ and $\sigma(p) \leftrightarrow F = T$.

(d) $\sigma = \{p = F, q = T, r = F\}$, since $\sigma(q \leftrightarrow r) = F$ and $\sigma(p) \leftrightarrow F = T$.

These are all the states that satisfy the requirement. Since p, q, r are proposition variables, there are only eight different possible states containing only p, q, r that are proper for $p \leftrightarrow q \leftrightarrow r$. If you list the other four states, you will see they do not satisfy $p \leftrightarrow q \leftrightarrow r$.

6. (a) False. The state is not proper for the predicate since b is an integer variable in the state but b represents an array in the predicate.
   (b) True. The expression can be evaluated in the given state, even though there will be a run-time error in during the evaluation.
   (c) True. T can be satisfied by any (well-formed) state.
   (d) True. $\sigma\big(p \leftrightarrow \big(\sigma(b[b[1]]) = 2\big)\big) = T \leftrightarrow \big(\sigma(b[0]) = 2\big) = T \leftrightarrow T = T$.
   (e) True. If a and b are the same array then $b[0]$ and $a[1][3]$ cannot be of the same type.

7. (a)

$$\begin{array}{lll} & \neg\forall x \geq 1.\ x^2 > x & \\ \equiv & \neg\forall x.\ x \geq 1 \rightarrow x^2 > x & \text{Definition of bounded quantifier} \\ \Leftrightarrow & \exists x.\ \neg(x \geq 1 \rightarrow x^2 > x) & \text{DeMorgan's Law} \\ \Leftrightarrow & \exists x.\ x \geq 1 \wedge x^2 \leq x & \text{Negation of} \rightarrow \end{array}$$

   (b)

$$\begin{array}{lll} & \neg\exists x.\exists y.\ x > y \wedge x < y & \\ \Leftrightarrow & \forall x.\ \neg\exists y.\ x > y \wedge x < y & \text{DeMorgan's Law} \\ \Leftrightarrow & \forall x.\forall y.\ \neg(x > y \wedge x < y) & \text{DeMorgan's Law} \\ \Leftrightarrow & \forall x.\forall y.\ x \leq y \vee x \geq y & \text{DeMorgan's Law} \end{array}$$

   (c)

$$\begin{array}{lll} & \neg\Big(\big(\exists x.\exists y.\ Q(x,y)\big) \wedge \forall x.\forall y.\ Q(y,x)\Big) & \\ \Leftrightarrow & \neg\big(\exists x.\exists y.\ Q(x,y)\big) \vee \neg\forall x.\forall y.\ Q(y,x) & \text{DeMorgan's Law} \\ \Leftrightarrow & \big(\forall x.\ \neg\exists y.\ Q(x,y)\big) \vee \exists x.\ \neg\forall y.\ Q(y,x) & \text{DeMorgan's Law} \\ \Leftrightarrow & \big(\forall x.\forall y.\ \neg Q(x,y)\big) \vee \exists x.\exists y.\ \neg Q(y,x) & \text{DeMorgan's Law} \end{array}$$

8. There are more than one correct ways to define these predicate functions.
   (a) $isGreater(b, m, x) \equiv 0 < m \leq size(b) \wedge \forall i.\ 0 \leq i < m \rightarrow x > b[i]$
   (b) $hasGreater(a, b) \equiv \forall 0 \leq i < size(b).\ \exists 0 \leq j < size(a).\ b[i] > a[j]$
   (c) $Extends(a, b) \equiv size(a) \leq size(b) \wedge \forall 0 \leq i < size(a).\ a[i] = b[i]$

9. Remind that, syntactic equality logically implies semantic equality.

		$\sigma[u \mapsto \alpha][v \mapsto \beta] = \sigma[v \mapsto \beta][u \mapsto \alpha]$?	$\sigma[u \mapsto \alpha][v \mapsto \beta] \equiv \sigma[v \mapsto \beta][u \mapsto \alpha]$?
$u \equiv v$	$\alpha = \beta$	Yes, because they are syntactically equal.	Yes, on both hand-sides we are updating the same variable with the same value twice.
$u \equiv v$	$\alpha \neq \beta$	No. On the left-hand-side, $u \equiv v$ is bind with β; and on the right hand side, $u \equiv v$ is bind with α.	No, because they are not semantically equal.
$u \not\equiv v$	$\alpha = \beta$	Yes. On both hand-sides, u and v are both bind with α (or β, since they are the same value).	No, they are not the same procedure since u and v are different variables.
$u \not\equiv v$	$\alpha \neq \beta$	Yes. On both hand-sides, u is bind with α and v is bind with β.	No, they are not the same procedure since u and v are different variables.

10. (a) $\sigma[x \mapsto \sigma(y)][y \mapsto \sigma(x)] = \sigma[x \mapsto 5][y \mapsto \sigma(x)] = \sigma[x \mapsto 5][y \mapsto 2] = \{x = 5, y = 2\}$.

    (b) $\gamma = \sigma[x \mapsto 3][y \mapsto \tau(x) * 4] = \sigma[x \mapsto 3]\big[y \mapsto \sigma[x \mapsto 3](x) * 4\big] = \sigma[x \mapsto 3][y \mapsto 12] = \{x = 3, y = 12\}$.