Logic Review; Expressions, States, Values, and Truth

CS 536: Science of Programming, Spring 2018

Due Thu Feb 1, 11:59 pm

2/4 Updates incorporated, solved; 2/9: pp. 4, 5

A. Why?

- Reviewing/studying logic is necessary because we'll be using it in the course.
- States describe memory; an expression has a value relative to a state.
- A predicate is satisfied relative to a state; it is valid if it is satisfied in all states.
- State updates occur when we introduce new variables or change the values of existing variables.

B. Objectives

At the end of this homework, you should be able to

- Describe the relationship between syntactic equality and semantic equality.
- Translate expressions, propositions, and predicates to and from English.
- Describe what a memory state is and how to read/write/update one.
- Calculate the value of an expression in a state; determine the satisfaction of a predicate in a state; describe what needs to be known in order to show that a predicate is valid
- Design predicate functions for simple properties on values and arrays.

C. Formatting and Submitting Your Work

- Submit your homework on Blackboard to the assignment folder for homework 1. It's probably best to do this as a pdf file. You can create the pdf by scanning hand-written answers or by generating one from your computer. You don't have to use a word processor to write out your answers: Feel free to convert logical symbols into ASCII text: For $\wedge$, $\vee$, $\rightarrow$, $\neg$, $\forall$, $\exists$,write `and`, `or`, `->`, `!`, `all`, and `exist`. For $\Rightarrow$, $\Leftrightarrow$, $\equiv$, and $\not\equiv$, write `=>`, `<=>`, `==`, and `!==`.
- You may work in groups of ≤ 4 people. Submit your answers just one time — Pick someone from your group and be sure to include the names and A-id numbers of the other group members.

D. Problems [100 points total]

Part 1: Logic Review

For these problems (actually, for basically the whole semester), p, q, and r are propositions. Quantified variables range over $\mathbb{Z}$ unless otherwise specified.

1. [6 = 3 * 2 points]
   a. Is $p \wedge q \vee \neg r \rightarrow \neg p \rightarrow q \equiv ((p \wedge q) \vee ((\neg r \rightarrow ((\neg p) \rightarrow q))))$?
   b. Is $(p \rightarrow ((\neg q) \rightarrow r \wedge (s \rightarrow p))) \equiv p \rightarrow \neg q \rightarrow r \wedge (s \rightarrow p)$?
   c. Is $\forall\, \mathtt{x} \,.\, p \rightarrow \exists\, \mathtt{y} \,.\, q \rightarrow r \equiv ((\forall\, \mathtt{x} \,.\, p) \rightarrow (\exists\, \mathtt{y} \,.\, q)) \rightarrow r$?

For Problems 2 and 3, Remember that because of associativity, we're treating $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r) \equiv p \wedge q \wedge r$. Because of left associativity, the full parenthesization of $p \wedge q \wedge r \equiv ((p \wedge q) \wedge r)$.

2. [6 = 2 * 3 points] Give the minimal parenthesization of each of the following by showing what remains after removing all redundant parentheses. Hint: In your scratch work, attach subscripts to the parentheses to match up $(_1 \ldots)_1$ versus $(_2$ and $)_2$ and so on.
   a. $(\neg(p \rightarrow (((\neg(q \wedge r)) \wedge ((\neg p \vee r) \vee (q \wedge s))))))$
   b. $\texttt{LboundsR}(\texttt{b}, \texttt{m}, \texttt{n}) \equiv (\exists\, \texttt{i}\, .\, (((0 \le \texttt{i}) \wedge (\texttt{i} < \texttt{m})) \wedge (\forall\, \texttt{j}\, .\, (((\texttt{m} \le \texttt{j}) \wedge (\texttt{j} < \texttt{n})) \rightarrow (\texttt{b[i]} = \texttt{b[j]})))))$.
   ($\texttt{LboundsR}(\texttt{b}, \texttt{m}, \texttt{n})$ asks "Is there a value in $\texttt{b[0..m-1]}$ > every value in $\texttt{b[m..n-1]}$?
3. [6 = 2 * 3 points] Give the full parenthesization of each of the following.
   (a) $p \wedge \neg r \wedge s \rightarrow \neg q \vee r \rightarrow \neg p \leftrightarrow \neg s \rightarrow t$
   (b) $\exists\, \texttt{m}\, .\, 0 \le \texttt{m} < \texttt{n} \wedge \forall\, \texttt{j}\, .\, 0 \le \texttt{j} < \texttt{m} \rightarrow \texttt{b[0]} \le \texttt{b[j]} \wedge \texttt{b[j]} \le \texttt{b[m]}$
4. [8 = 4 * 2 points] Say whether each of the following is a tautology, contradiction, or contingency. (Assume `s`, `t`, and `u` are boolean variables; assume $p(\texttt{x}, \texttt{y})$ is a predicate function.)
   a. $\texttt{s} \wedge \texttt{t} \vee \texttt{u} \rightarrow \texttt{s} \vee \texttt{u}$
   b. $\neg(\texttt{s} \rightarrow \texttt{t}) \leftrightarrow (\neg \texttt{s} \vee \texttt{t})$
   c. $(\forall\, \texttt{x}\, .\, \forall\, \texttt{y}\, .\, p(\texttt{x}, \texttt{y})) \leftrightarrow (\forall\, \texttt{y}\, .\, \forall\, \texttt{x}\, .\, p(\texttt{x}, \texttt{y}))$
   d. $(\exists\, \texttt{x}\, .\, \exists\, \texttt{y}\, .\, p(\texttt{x}, \texttt{y})) \leftrightarrow (\exists\, \texttt{y}\, .\, \exists\, \texttt{x}\, .\, p(\texttt{x}, \texttt{y}))$
5. [4 points] Which of the following mean $p \rightarrow q$ and which mean $q \rightarrow p$?
   a. p is sufficient for q
   b. p only if q
   c. $p \leftarrow q$
   d p is necessary for q
6. [6 = 3 * 2 points] For each of the following possible definitions for $p(u, v)$, (1) Is $(\forall\, \texttt{x}\, .\, \exists\, \texttt{y}\, .\, p(\texttt{x}, \texttt{y}))$ valid? (2) Is $(\exists\, \texttt{y}\, .\, \forall\, \texttt{x}\, .\, p(\texttt{x}, \texttt{y}))$ valid? (3) Is $(\forall\, \texttt{x}\, .\, \exists\, \texttt{y}\, .\, p(\texttt{x}, \texttt{y}))$ is $\rightarrow$, $\leftarrow$, or $\leftrightarrow$ to $(\exists\, \texttt{y}\, .\, \forall\, \texttt{x}\, .\, p(\texttt{x}, \texttt{y}))$? (Hint: say which of F $\leftrightarrow$ F, F $\rightarrow$ T, T $\leftarrow$ F, or T $\leftrightarrow$ T holds.)
   a. $p(u, v) \equiv v \le u^2$
   b. $p(u, v) \equiv u > v$
   c. $p(u, v) \equiv 0 < v < u^2$
7. [6 = 3 * 2 points] Let e_1 and e_2 be expressions that rely only on `x` (no other variables). For each statement below, say whether the statement is correct or incorrect (and if incorrect, give definitions for e_1 and e_2 that form a counterexample).
   a. If $e_1 \equiv e_2$ then $(\forall\, \texttt{x}\, .\, e_1 = e_2)$ is true
   b. If $(\forall\, \texttt{x}\, .\, e_1 = e_2)$ then $e_1 \equiv e_2$ is true
   c. If $e_1 \not\equiv e_2$ then $(\exists\, \texttt{x}\, .\, e_1 \ne e_2)$ is true
8. [8 points] Simplify $\neg(\forall\, \texttt{x}\, .\, (\exists\, \texttt{y}\, .\, \texttt{x} \le \texttt{y}) \vee \exists\, \texttt{z}\, .\, \texttt{x} \ge \texttt{z})$ to a predicate that has no uses of $\neg$. (You'll need DeMorgan's laws.) Present a proof of equivalence using format of the example below.

$\neg(\texttt{x} < \texttt{y} \wedge \texttt{y} \le \texttt{z})$	
$\Leftrightarrow \neg(\texttt{x} < \texttt{y}) \vee \neg(\texttt{y} \le \texttt{z})$	DeMorgan's Law
$\Leftrightarrow \texttt{x} \ge \texttt{y} \vee \texttt{y} > \texttt{z}$	Negation of comparison (twice)

   (Be sure to include the names of the rules you're applying!)

Part 2: Expressions, States, Satisfaction, and Validity

9. [6 = 3 * 2 points] Which of the following expressions are legal or illegal according to the syntax we're using? Assume x, y, z are integer variables, f is function, and b and b′ are array names.
   a. (x < y ? T : 17)
   b. (i = 3 ? b[1] : b′[2])[k]
   c. f(b, b′, x)
10. [6 = 3 * 2 points] For each of the following, say whether the state is legal (and if not, why not). If the state is legal, then is it proper for the expression (and if not, why not). If the state is proper, then what is the value of the expression in that state?
    a. {x = α, y = 2+x} and x+y
    b. {b = γ, i = 1} where γ(0) = 2 and γ(1) = 5, and b[b[i]-1]]
    c. {b = the function that maps 0 to 2 and 1 to 5, i = one} and b+i
    d. {c = α, d = 2α, e = 3α} (for some α} and d/c+(0*z)
    e. {b = (5), i = 0, x = 1} and x > b[i]. (Hint: are the bindings b = (5) and b = 5 different?)
    f. {x = 8, y = 1} ∪ {x = 9} and 3*y
    g. {x = T, y = 1} and (y > 0 ? x : x+y)i
11. [8 = 4 * 2 points] Let σ = {x = 2, y = 4, b = (11, 21, 31, 41)}.
    a. What is σ[x ↦ 8][x ↦ 5]?
    b. What is σ[y ↦ 5](x)?
    c. What is σ[b[1] ↦ 13][y ↦ σ(b[1]/2)]? (Assumes integer division truncates, as in C.)
    d. When does σ[u ↦ α][v ↦ β] = σ[v ↦ β][u ↦ α]? (Note u ≡ or ≢ v and α = or ≠ β.)
12. [12 = 4 * 3 points] Let σ(x) = 2 and σ(y) be undefined. (So σ may or may not be defined on other variables.) Which of the following are correct and if not, why not? (Is the state illegal or improper? Or is the state proper but just doesn't satisfy the predicate?)
    a. σ[y ↦ 5] ⊨ ∃ x . x > y
    b. σ[x ↦ 3] ⊨ ∀ y . y ≥ 2 → x < y^2
    c. σ[y ↦ 2] ⊨ x = y
    d. σ[z ↦ 5][y ↦ 3][z ↦ 6] ⊨ z = x*y
13. [12 = 4 * 3 points] Using the same σ as in the previous problem, which of the following involve legal states, and is the claim true? (If one or more states are illegal, say why.)
    a. σ[z ↦ 4] = σ ∪ {z = 4}
    b. σ[y ↦ 0][b ↦ (1, 3, σ(x+y))]
    c. σ[y ↦ 0][b ↦ (1, 3, σ[y ↦ 0](x+y))]
    d. σ[v ↦ 5](x) is undefined
14. [3 points] Describe in English when ⊭ (∃ x . ∀ y . x < f(y)). (For some? every? state where …)
15. [3 points] Show ⊭ ∀x ∈ ℤ . x > 0 → x^2 > x by giving a value for x that provides a counterexample. What property does this state need to have? (I.e., does it ⊨ or ⊭ something?)

Solution to Homework 1 — Logic Review; Expressions, States, Values, and Truth

Spring 2018

Answer

1. (Syntactic equality) For the ones that are $\not\equiv$, I'm also giving the minimal parenthesization too (you didn't have to do that).

   1a. No. $p \wedge q \vee \neg r \rightarrow \neg p \rightarrow q \not\equiv ((p \wedge q) \vee ((\neg r \rightarrow ((\neg p) \rightarrow q)))) \equiv p \wedge q \vee (\neg r \rightarrow \neg p \rightarrow q)$, however.

   1b. Yes, $(p \rightarrow ((\neg q) \rightarrow r \wedge (s \rightarrow p))) \equiv p \rightarrow \neg q \rightarrow r \wedge (s \rightarrow p)$

   1c. No, $\forall\, \mathtt{x} \,.\, p \rightarrow \exists\, \mathtt{y} \,.\, q \rightarrow r \not\equiv ((\forall\, \mathtt{x} \,.\, p) \rightarrow (\exists\, \mathtt{y} \,.\, q)) \rightarrow r \equiv (\forall\, \mathtt{x} \,.\, p) \rightarrow (\exists\, \mathtt{y} \,.\, q) \rightarrow r$, however.

2. (Minimal parenthesization) To make it clear what matches what, I've included subscripts on the parentheses we kept. (You didn't have to do that.)

   2a. $(\neg (_1\, p \rightarrow (((_2 \neg (q \wedge r))_2 \wedge (_3 (\neg p \vee r) \vee (q \wedge s))_3)))_1) \equiv \neg (p \rightarrow \neg (q \wedge r) \wedge (\neg p \vee r \vee q \wedge s))$

   2b. $\mathtt{LboundsR}(\mathtt{b}, \mathtt{m}, \mathtt{n})$

   $\equiv (\exists\, \mathtt{i} \,.\, (((0 \le \mathtt{i}) \wedge (\mathtt{i} < \mathtt{m})) \wedge (\forall\, \mathtt{j} \,.\, (((\mathtt{m} \le \mathtt{j}) \wedge (\mathtt{j} < \mathtt{n})) \rightarrow (\mathtt{b[i]} = \mathtt{b[j]})))))$

   $\equiv \exists\, \mathtt{i} \,.\, 0 \le \mathtt{i} \wedge \mathtt{i} < \mathtt{m} \wedge \forall\, \mathtt{j} \,.\, \mathtt{m} \le \mathtt{j} \wedge \mathtt{j} < \mathtt{n} \rightarrow \mathtt{b[i]} = \mathtt{b[j]}$

3. (Full parenthesization) To make it clearer what matches what, I've included subscripts (you didn't have to do that.)

   3a. $p \wedge \neg r \wedge s \rightarrow \neg q \vee r \rightarrow \neg p \leftrightarrow \neg s \rightarrow t$

   ~~$\equiv (_1 (_2 (_3\, p \wedge (_4 \neg r)_4)_3 \wedge s)_2 \rightarrow (_5 (_6 (_7 (_8 \neg q)_8 \vee r)_7 \rightarrow (_9 \neg p)_9)_6 \leftrightarrow (_{10} (_{11} \neg s)_{11} \rightarrow t)_5)_1$~~ [2/9]

   $\equiv (_1 (_2 (_3 (_4\, p \wedge (_5 \neg r)_5)_4 \wedge s)_3 \rightarrow (_6 (_7 (_8 \neg q)_8 \vee r)_7 \rightarrow (_9 \neg p)_9)_6)_2 \leftrightarrow (_{10} (_{11} \neg s)_{11} \rightarrow t)_{10})_1$

   3b. $\exists\, \mathtt{m} \,.\, 0 \le \mathtt{m} < \mathtt{n} \wedge \forall\, \mathtt{j} \,.\, 0 \le \mathtt{j} < \mathtt{m} \rightarrow \mathtt{b[0]} \le \mathtt{b[j]} \wedge \mathtt{b[j]} \le \mathtt{b[m]}$

   $\equiv (_1\, \exists\, \mathtt{m} \,.\, (_2 (_3\, 0 \le \mathtt{m} < \mathtt{n})_3$

   $\wedge (_4\, \forall\, \mathtt{j} \,.\, (_5 (_6\, 0 \le \mathtt{j} < \mathtt{m})_6 \rightarrow (_7 (_8\, \mathtt{b[0]} \le \mathtt{b[j]})_8 \wedge (_9\, \mathtt{b[j]} \le \mathtt{b[m]})_9)_7)_5)_4)_2)_1$

   (You can break down $(0 \le \mathtt{j} < \mathtt{m})$ into $((0 \le \mathtt{j}) \wedge (\mathtt{j} < \mathtt{m}))$, but it wasn't an explicit requirement.)

4. (Tautologies, contradictions, and contingencies)

   a. Tautology: $\mathtt{s} \wedge \mathtt{t} \vee \mathtt{u} \rightarrow \mathtt{s} \vee \mathtt{u}$

   b. [2/9] Contradiction ~~Tautology~~: $\neg(\mathtt{s} \rightarrow \mathtt{t}) \leftrightarrow (\neg \mathtt{s} \vee \mathtt{t})$

   c. Tautology: $(\forall\, \mathtt{x} \,.\, \forall\, \mathtt{y} \,.\, p(\mathtt{x}, \mathtt{y})) \leftrightarrow (\forall\, \mathtt{y} \,.\, \forall\, \mathtt{x} \,.\, p(\mathtt{x}, \mathtt{y}))$

   d. Tautology: $(\exists\, \mathtt{x} \,.\, \exists\, \mathtt{y} \,.\, p(\mathtt{x}, \mathtt{y})) \leftrightarrow (\exists\, \mathtt{y} \,.\, \exists\, \mathtt{x} \,.\, p(\mathtt{x}, \mathtt{y}))$

   (Gee, that was kind of boring :-)

5. ($\rightarrow$ vs $\leftarrow$)

   a. $p \rightarrow q$: p is sufficient for q

   b. $p \rightarrow q$: p only if q

   c. $q \rightarrow p$: $p \leftarrow q$

   d $q \rightarrow p$: p is necessary for q

6. (∀ ∃ vs ∃ ∀)

	Defn $p(u, v)$	$\forall u . \exists v . p(u, v)$	$\exists v . \forall u . p(u, v)$	(∀ ∃) relation (∃ ∀)?
a.	$p(u, v) \equiv v \le u^2$	Valid	Valid	$\leftrightarrow$
b.	$p(u, v) \equiv u > v$	Valid *	Invalid	$\leftarrow$ [2/9]
c.	$p(u, v) \equiv 0 < v < u^2$	Invalid	Invalid	$\leftrightarrow$

(*Remember, $u, v \in \mathbb{Z}$. Were $u, v \in \mathbb{N}$, the answer would be different.)

7. (≡ vs =)

a. If $e_1 \equiv e_2$ then $(\forall x . e_1 = e_2)$ is true — Correct

b. If $(\forall x . e_1 = e_2)$ then $e_1 \equiv e_2$ is true — Incorrect: $2+2 = 4$ but $2+2 \not\equiv 4$

c. If $e_1 \not\equiv e_2$ then $(\exists x . e_1 \ne e_2)$ is true — Incorrect: $2+2 \not\equiv 4$ but $2+2 = 4$

8. (Remove ¬)

$\neg(\forall x . (\exists y . x \le y) \vee \exists z . x \ge z)$

$\Leftrightarrow \exists x . \neg((\exists y . x \le y) \vee \exists z . x \ge z)$ — DeMorgan's Law ($\neg\forall \Leftrightarrow \exists\neg$)

$\Leftrightarrow \exists x . \neg(\exists y . x \le y) \wedge \neg\exists z . x \ge z$ — DeMorgan's Law ($\neg\vee \Leftrightarrow \neg\wedge\neg$)

$\Leftrightarrow \exists x . (\forall y . x > y) \wedge \neg\exists z . x \ge z$ — DeMorgan's Law ($\neg\exists \Leftrightarrow \forall\neg$) and ¬ of ≤

$\Leftrightarrow \exists x . (\forall y . x > y) \wedge \forall z . x < z$ — DeMorgan's Law ($\neg\exists \Leftrightarrow \forall\neg$) and ¬ of ≥

Part 2: Expressions, States, Satisfaction, and Validity

9. (Legal expressions) I included some comments; you weren't required to do that.

a. `(x < y ? T : 17)` — Illegal: (`T` and `17` aren't the same type)

b. `(i = 3 ? b[1] : b′[2])[k]` — Illegal: (Conditional can't yield an array)

c. `f(b, b′, x)` — Legal: (`f` takes two arrays and an integer)

10. (Legal and proper states, value of an expression) [6 = 3 * 2 points] For each of the following, say whether the state is legal (and if not, why not). If the state is legal, then is it proper for the expression (and if not, why not). If the state is proper, then what is the value of the expression in that state?

10a. {`x` = α, `y` = 2+`x`} and `x+y`

State is illegal (`x` can't be a variable and a value simultaneously)

10b. {`b` = γ, `i` = 1} where γ(0) = 2 and γ(1) = 5, and `b[b[i]-1]]`

State is legal and proper, value of `b[b[i]-1]]` is γ(γ(1)−1) = γ(5−1) = γ(4) is out of range.

10c. {`b` = the function that maps 0 to 2 and 1 to 5, `i` = one} and `b+i`

Legal state but improper: You can't add a function and an integer.

(An example of a legal and proper state is: {`b` = 3, `i` = 1}.)

10d. {c = α, d = 2α, e = 3α} (for some α} and `d/c+(0*z)`

Legal but improper state (no value for `z`; if `z` had a value, the value of `d/c+(0*z)` would be 2).

10e. $\{\mathtt{b} = (5), \mathtt{i} = 0, \mathtt{x} = 1\}$ and $\mathtt{x} > \mathtt{b[i]}$. (Hint: are the bindings $\mathtt{b} = (5)$ and $\mathtt{b} = 5$ different?)

Legal and proper state; value of $\mathtt{b[i]}$ is the function (5) applied to 0, which is 5, so the value of $\mathtt{x} > \mathtt{b[i]}$ is $1 > 5$, which is false.

10f. $\{\mathtt{x} = 8, \mathtt{y} = 1\} \cup \{\mathtt{x} = 9\}$ and $\mathtt{3*y}$

Illegal: Has two bindings for $\mathtt{x}$.

10g. $\{\mathtt{x} = \mathrm{T}, \mathtt{y} = 1\}$ and $(\mathtt{y} > \mathtt{0}\ \mathtt{?}\ \mathtt{x}\ \mathtt{:}\ \mathtt{x+y})$

Legal state but improper: We need $\mathtt{x}$ to be an integer.

11. (State updates. Below, $\sigma = \{\mathtt{x} = 2, \mathtt{y} = 4, \mathtt{b} = \gamma\}$ where $\gamma = (11, 21, 31, 41)$.

11a. $\sigma[\mathtt{x} \mapsto 8][\mathtt{x} \mapsto 5] = \{\mathtt{x} = 2, \mathtt{y} = 4, \mathtt{b} = \gamma\}[\mathtt{x} \mapsto 8][\mathtt{x} \mapsto 5] = \{\mathtt{x} = 8, \mathtt{y} = 4, \mathtt{b} = \gamma\}[\mathtt{x} \mapsto 5]$
$= \{\mathtt{x} = 5, \mathtt{y} = 4, \mathtt{b} = \gamma\}$

11b. $\sigma[\mathtt{y} \mapsto 5](\mathtt{x}) = \{\mathtt{x} = 2, \mathtt{y} = 4, \mathtt{b} = \gamma\}[\mathtt{y} \mapsto 5](\mathtt{x}) = \{\mathtt{x} = 2, \mathtt{y} = 5, \mathtt{b} = \gamma\}(\mathtt{x}) = 2$

11c. For $\sigma[\mathtt{b[1]} \mapsto 13][\mathtt{y} \mapsto \sigma(\mathtt{b[1]/2})]$, note that we're taking the value of $\mathtt{b[1]/2}$ in σ, not $\sigma[\mathtt{b[1]} \mapsto 13]$. If you want the value of $\mathtt{b[1]/2}$ in that state, you have to write $\sigma[\mathtt{b[1]} \mapsto 13](\mathtt{b[1]/2})$.

Since $\sigma(\mathtt{b[1]/2}) = \sigma(\mathtt{b})(1) / 2 = 21/2 = 10$, we get

$\sigma[\mathtt{b[1]} \mapsto 13][\mathtt{y} \mapsto \sigma(\mathtt{b[1]/2})] = \sigma[\mathtt{b[1]} \mapsto 13][\mathtt{y} \mapsto 10]$
$= \{\mathtt{x} = 2, \mathtt{y} = 4, \mathtt{b} = \gamma\}[\mathtt{b[1]} \mapsto 13][\mathtt{y} \mapsto 10]$
$= \{\mathtt{x} = 2, \mathtt{y} = 4, \mathtt{b} = \gamma[1 \mapsto 13]\}[\mathtt{y} \mapsto 10]$ where $\gamma = (11, 21, 31, 41)$
$= \{\mathtt{x} = 2, \mathtt{y} = 4, \mathtt{b} = (11, 21, 31, 41)[1 \mapsto 13]\}[\mathtt{y} \mapsto 10]$
$= \{\mathtt{x} = 2, \mathtt{y} = 4, \mathtt{b} = (11, 13, 31, 41)\}[\mathtt{y} \mapsto 10]$
$= \{\mathtt{x} = 2, \mathtt{y} = 10, \mathtt{b} = (11, 13, 31, 41)\}$

11d. (When does $\sigma[u \mapsto \alpha][v \mapsto \beta] = \sigma[v \mapsto \beta][u \mapsto \alpha]$?)

Case 1 ($u \equiv v$) On the left-hand side, we find $\sigma[u \mapsto \alpha][v \mapsto \beta] = \sigma[v \mapsto \beta]$. On the right-hand side, we find $\sigma[v \mapsto \beta][u \mapsto \alpha] = \sigma[u \mapsto \alpha] = \sigma[v \mapsto \alpha]$ (since $u \equiv v$). The two sides are $\equiv$ when $\sigma[v \mapsto \beta] = \sigma[v \mapsto \alpha]$, which happens exactly when $\alpha = \beta$.

Case 2 ($u \not\equiv v$) In this case, the two sides are equal: updating v after updating u won't affect the binding for u and vice versa, so it doesn't matter whether we update u then v or v then u.

Summary: The two are equal when ($u \equiv v$ and $\alpha = \beta$) or ($u \not\equiv v$).

12. (Satisfaction of predicates) Recall $\sigma(\mathtt{x}) = 2$ and $\sigma(\mathtt{y})$ may be undefined.

12a. $\sigma[\mathtt{y} \mapsto 5] \vDash \exists\, \mathtt{x}\ \mathtt{.}\ \mathtt{x} > \mathtt{y}$ with 6 as the witness: $\sigma[\mathtt{y} \mapsto 5] \vDash \exists\, \mathtt{x}\ \mathtt{.}\ \mathtt{x} > \mathtt{y}$ if $\sigma[\mathtt{y} \mapsto 5][\mathtt{x} \mapsto 6] \vDash \mathtt{x} > \mathtt{y}$ iff $6 > 5$, which is true.

12b. First, even if $\sigma(\mathtt{y})$ is undefined, $\sigma[\mathtt{x} \mapsto 3]$ is proper for $(\forall\, \mathtt{y} \ldots)$ because the quantifier makes the status of $\sigma(\mathtt{y})$ irrelevant. From the definition of $\vDash$ for $\forall$, $\sigma[\mathtt{x} \mapsto 3] \vDash \forall\, \mathtt{y}\ \mathtt{.}\ \mathtt{y} \geq \mathtt{2} \rightarrow \mathtt{x} < \mathtt{y}^2$ iff for all $\alpha \in \mathbb{Z}$, $\sigma[\mathtt{x} \mapsto 3][\mathtt{y} \mapsto \alpha] \vDash \mathtt{y} \geq \mathtt{2} \rightarrow \mathtt{x} < \mathtt{y}^2$. If $\alpha < 2$, then the implication is satisfied (because false implies anything). If $\alpha \geq 2$, then $\alpha^2 \geq 2^2 = 4$, so $\sigma[\mathtt{x} \mapsto 3][\mathtt{y} \mapsto \alpha] \vDash \mathtt{y} \geq \mathtt{2} \rightarrow \mathtt{x} < \mathtt{y}^2$ because in $\sigma[\mathtt{x} \mapsto 3][\mathtt{y} \mapsto \alpha]$, the value of $\mathtt{y} \geq \mathtt{2}$ is true and (the value of $\mathtt{x} < \mathtt{y}^2$) = the value of $(3 < 4 \leq \alpha^2)$ = true.

12c. $\sigma[\mathtt{y} \mapsto 2] \vDash \mathtt{x} = \mathtt{y}$ is true: $\sigma[\mathtt{y} \mapsto 2](\mathtt{x}) = \sigma(\mathtt{x})$ [since $\mathtt{x} \not\equiv \mathtt{y}$] $= 2$ by assumption, and $\sigma[\mathtt{y} \mapsto 2](\mathtt{y}) = 2$.

12d. $\sigma[\mathtt{z} \mapsto 5][\mathtt{y} \mapsto 3][\mathtt{z} \mapsto 6] \vDash \mathtt{z} = \mathtt{x*y}$ is true. The second update of z overwrites the first update, so $\sigma[\mathtt{z} \mapsto 5][\mathtt{y} \mapsto 3][\mathtt{z} \mapsto 6] = \sigma[\mathtt{y} \mapsto 3][\mathtt{z} \mapsto 6]$. Since x, y, and z are pairwise $\not\equiv$, $\sigma[\mathtt{y} \mapsto 3][\mathtt{z} \mapsto 6](\mathtt{x}) = \sigma[\mathtt{y} \mapsto 3](\mathtt{x}) = \sigma(\mathtt{x})$, which is 2 by assumption. Because of the updates, $\sigma[\mathtt{y} \mapsto 3][\mathtt{z} \mapsto 6](\mathtt{y}) = \sigma[\mathtt{y} \mapsto 3](\mathtt{y}) = 3$ and $\sigma[\mathtt{y} \mapsto 3][\mathtt{z} \mapsto 6](\mathtt{z}) = 6$. So $\sigma[\mathtt{y} \mapsto 3][\mathtt{z} \mapsto 6] \vDash \mathtt{z} = \mathtt{x*y}$ iff $6 = 2 \times 3$, which is true.

13. (Updated states) As in the previous problem, $\sigma(\mathtt{x}) = 2$ and $\sigma(\mathtt{y})$ may be undefined.

[12 = 4 * 3 points] Using the same σ as in the previous problem, which of the following involve legal states, and is the claim true? (If one or more states are illegal, say why.)

13a. $\sigma[\mathtt{z} \mapsto 4] = \sigma \cup \{\mathtt{z} = 4\}$ — This is legal iff $\sigma(\mathtt{x})$ is undefined; from our assumption, it may or may not.

13b. $\sigma[\mathtt{y} \mapsto 0][\mathtt{b} \mapsto (1, 3, \sigma(\mathtt{x+y}))]$ is not legal. We're given that $\sigma(\mathtt{y})$ may be undefined, in which case σ is improper for x+y.

c. $\sigma[\mathtt{y} \mapsto 0][\mathtt{b} \mapsto (1, 3, \sigma[\mathtt{y} \mapsto 0](\mathtt{x+y}))]$ — This is legal: $\sigma[\mathtt{y} \mapsto 0](\mathtt{x+y}) = \sigma[\mathtt{y} \mapsto 0](\mathtt{x}) + \sigma[\mathtt{y} \mapsto 0](\mathtt{y}) = \sigma(\mathtt{x}) + 0 = 2$, so in effect, we have $\sigma[\mathtt{y} \mapsto 0][\mathtt{b} \mapsto (1, 3, 2)]$, which is more obviously correct.

d. $\sigma[\mathtt{v} \mapsto 5](\mathtt{x})$ is undefined — This is false: Since $\mathtt{v} \not\equiv \mathtt{x}$, $\sigma[\mathtt{v} \mapsto 5](\mathtt{x}) = \sigma(\mathtt{x}) = 2$.

14. (English description of when $\nvDash \exists\, \forall$) Below, the key ideas are that

(1) Invalidity ($\nvDash p$) means unsatisfied in some state: For some σ, $\sigma \nvDash p$

(2) $\sigma \nvDash \exists\, \mathtt{x} \dots$ means there's no witness value for x, and

(3) $\sigma \nvDash \forall\, \mathtt{y} \dots$ means that for some value of y, the body is not satisfied.

With all that in mind, we get $\nvDash (\exists\, \mathtt{x} \,.\, \forall\, \mathtt{y} \,.\, \mathtt{x} < \mathtt{f(y)})$

iff for some σ, $\sigma \nvDash (\exists\, \mathtt{x} \,.\, \forall\, \mathtt{y} \,.\, \mathtt{x} < \mathtt{f(y)})$

iff for some σ, for all α, $\sigma[\mathtt{x} \mapsto \alpha] \nvDash \forall\, \mathtt{y} \,.\, \mathtt{x} < \mathtt{f(y)}$

iff for some σ, for all α, for some β, $\sigma[\mathtt{x} \mapsto \alpha][\mathtt{y} \mapsto \beta] \nvDash \mathtt{x} < \mathtt{f(y)}$

15. (Case showing $\nvDash \forall \mathtt{x} \in \mathbb{Z} \,.\, \mathtt{x} > 0 \rightarrow \mathtt{x}^2 > \mathtt{x}$). Again, ($\nvDash p$) means (for some σ, $\sigma \nvDash p$), and ($\sigma \nvDash \forall\, \mathtt{x} \,.\, \mathit{body}$) means for some value β for x, $\sigma[\mathtt{x} \mapsto \beta] \nvDash \mathit{body}$. The value β is what I meant by a counterexample for ($\nvDash \forall \mathtt{x} \in \mathbb{Z} \,.\, \mathtt{x} > 0 \rightarrow \mathtt{x}^2 > \mathtt{x}$). For this predicate, β = 1 works: $\nvDash \forall \mathtt{x} \in \mathbb{Z} \,.\, \mathtt{x} > 0 \rightarrow \mathtt{x}^2 > \mathtt{x}$

iff for some σ, $\sigma \nvDash \forall \mathtt{x} \in \mathbb{Z} \,.\, \mathtt{x} > 0 \rightarrow \mathtt{x}^2 > \mathtt{x}$

iff for some β, $\sigma[\mathtt{x} \mapsto \beta] \nvDash \mathtt{x} > 0 \rightarrow \mathtt{x}^2 > \mathtt{x}$

which holds when β = 1: $\sigma[\mathtt{x} \mapsto 1] \nvDash \mathtt{x} > 0 \rightarrow \mathtt{x}^2 > \mathtt{x}$

So the counterexample value is 1, and we need $\sigma[\mathtt{x} \mapsto 1] \nvDash \mathtt{x} > 0 \rightarrow \mathtt{x}^2 > \mathtt{x}$ to show this.