Assignment 1

Wednesday, January 17, 2024 4:32 PM

Part A) Statements in Logic Notation

Do Questions 3.7, 3.8, and 3.9 in the textbook (page 93) (6 points)

3.7 `x * y` is a list if `x * y` is valid Python and `x` and `y` are not both numeric values.
3.8 if `x + y` is a list, then `x * y` is not a list.
3.9 `x + y` and `x ** y` are both valid Python only if `x` is not a list.

3.7. Let

p :	`x + y` is valid Python	u :	x is a numeric value
q :	`x * y` is valid Python	v :	y is a numeric value
r :	`x ** y` is valid Python	w :	x is a list
s :	`x * y` is a list	z :	y is a list
t :	`x + y` is a list		

Then, the compound prop is as follows:

$q \wedge \neg(u \wedge v) \Rightarrow S$

3.8 Let

p :	`x + y` is valid Python	u :	x is a numeric value
q :	`x * y` is valid Python	v :	y is a numeric value
r :	`x ** y` is valid Python	w :	x is a list
s :	`x * y` is a list	z :	y is a list
t :	`x + y` is a list		

Then, the compound prop is as follows:

$t \Rightarrow \neg S$

3.9. Let

p :	`x + y` is valid Python	u :	x is a numeric value
q :	`x * y` is valid Python	v :	y is a numeric value
r :	`x ** y` is valid Python	w :	x is a list
s :	`x * y` is a list	z :	y is a list
t :	`x + y` is a list		

Then, the compound prop is as follows:

$(P \wedge r) \Leftrightarrow \neg w$

Part B) Truth Tables, Logic Equivalence, and Logic Laws

B.1) Show using truth tables that the following expression is a tautology (8 points)

$$(p \Rightarrow q) \wedge \neg q \Rightarrow \neg p$$

P	q	¬P	¬q	P ⇒ q	(P ⇒ q) ∧ ¬q
T	T	F	F	T	F
T	F	F	T	F	F

F	T	T	F	T	F
F	F	T	T	T	T

$(p \Rightarrow q) \wedge \neg q \Rightarrow \neg p$

T

T

T

T

B.2) Prove, using the logic equivalences discussed in class, that a conditional proposition is logically equivalent to its contrapositive. (8 points)

Proof. Let p and q be some statement,

Then we want to show that:

$p \Rightarrow q \equiv \neg q \Rightarrow \neg p$

WKT: $p \Rightarrow q \equiv \neg p \vee q$ by def.

$\equiv \neg(p \wedge \neg q)$ by De Morgan's

$\equiv \neg(\neg q \wedge p)$ by Commutative

$\equiv \neg(\neg q) \vee \neg p$ by De Morgan's

$\equiv \neg q \Rightarrow \neg p$ by def.

$\square$

∴ By using demorg's, commutative and def of implication, we learned that $p \Rightarrow q \equiv \neg q \Rightarrow \neg p$.

Part C) Logic and Programming

Do question 3.56 in the textbook (Page 106) (8 points)

3.56 Simplify the code in Figure 3.17a as much as possible. (For example, if $p \Rightarrow q$, it's a waste of time to test whether q holds in a block where p is known to be true.)

(a)

```
if x > 20 or (x ≤ 20 and y < 0) then
    foo(x, y)
else
    bar(x, y)
```

Let p: $x > 20$

q: $x \leq 20$

r: $y < 0$

then, the compound prop is,

$p \vee (q \wedge r) \Rightarrow foo(x, y)$

$\neg(p \vee (q \wedge r) \Rightarrow bar(x, y)$