## 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 t: x + y is a list

Then, the compound prop is as follows:  $9 \wedge 7(u \wedge v) = 5$ 

Then, the compaind prop is as follows:

3. Q. 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 t: x + y is a list

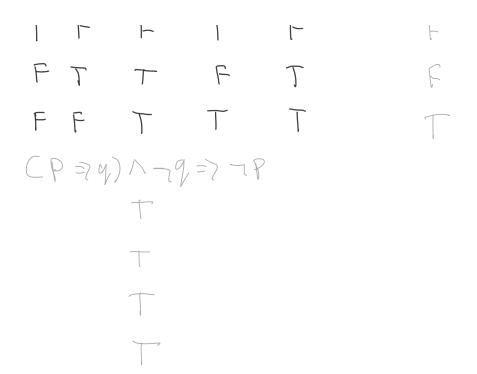
Then, He compound prop is as follows:

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

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

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

P q  $\neg P$   $\neg q$   $P \Rightarrow q$   $(P \Rightarrow q) \land \neg q$  T T F F 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:

 $P \Rightarrow q \equiv \neg q \Rightarrow \neg P$ 
 $WKT = P \Rightarrow q \equiv \neg P \lor q$  by def.

 $E = \neg (P \land \neg q)$  by De Morgan's

 $E = \neg (\neg q \land P)$  by Commutative

 $E = \neg (\neg q) \lor \neg P$  by De Morgan's

 $E = \neg (\neg q) \lor \neg P$  by def.

:- By using demorg's, commutative and def of implication, we learned that  $P=q\equiv \neg q=>\neg P$ .

## Part C) Logic and Programming

**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) 1 **if** x > 20 or  $(x \le 20 \text{ and } y < 0)$  **then**2 foo(x, y)3 **else**4 bar(x, y)
- Let P: x>20

9: X £ 20

r: 440

then, the compaind prop is,  $PV(q, \Lambda r) \Rightarrow foo(x, y)$ 

-(PV(q NT) => bar(x,y)