Classical Conversations ~ Challenge B

Intermediate Logic Appendices

"Self-Talk" while copying

Appendix A ~ Defining Truth Tables for the Five Logical Operators

Learning just ONE key for each table gives you the entire table (yay!):

Negation:	negation always gives the opposite truth value
Conjunction:	the only way a conjunction can be true is when both p and q are true
Disjunction:	the only way a disjunction can be false is when both p and q are false
Conditional:	the only way a conditional can be false is when p is true and q is false
Biconditional:	a biconditional is true whenever the truth values of p and q match

Quicker way to recreate Appendix A:

Negation				Conjunction	Disjunction	Conditional	Biconditional
p	~p	p	q	p • q	p ∨ q	p ⊃ q	p ≡ q
T	F	T	T	**T**	T	T	**T**
F	T	T	F	F	T	**F**	F
		F	T	F	T	T	F
		F	F	F	**F**	T	**T**

Appendix B ~ Rules of Replacement

10. De Morgan's Theorems:
    - 1st line: conjunction is false whenever either statement is false (see AppA)
    - 2nd line: disjunction is false only when both statements are false (see AppA)
11. Commutation (= commutative law): for • or ∨, order doesn't matter
12. Association (= associative law): for • or ∨, grouping doesn't matter
13. Distribution (= distributive law but works both ways): distribute • over ∨ or ∨ over •
14. Double Negation (= double negative): 2 negatives makes a positive

15. Transposition: when the conditional is true, then if q is false, p must be false
    OR: when the conditional is true, you can switch the stmts if you negate both
    (like contrapositive in 1st semester)
16. Material Impl: when the conditional is true, either p is false or q is true (or both)
17. Material Equivalence:
    - 1st line: when the biconditional is true, then the conditional is true in both directions (p ⊃ q and q ⊃ p) – from the definition of biconditional
    - 2nd line: when the biconditional is true, then the truth values of p and q MATCH-- EITHER p and q are both true OR p and q are both false – from the truth table
18. Exportation: when the truth of both p and q together makes r true, then you can make a chain of conditionals (not very helpful, I know; but neither is the description in the book; I don't remember it being used very often so don't worry about it ☺)
19. Tautology:
    - 1st line: when p is true then p ∨ p is true (because disj. is true when both parts are true)
    - 2nd line: when p is true then p • p is true (because conj. is true when both parts are true)

Classical Conversations ~ Challenge B
Intermediate Logic Appendices
"Self-Talk" while copying ~ continued

Appendix C ~ Decomposition Rules

NOTES before you begin:
Appendix A starts with what you know about the truth values of PARTS of a compound statement and tells you what you can figure out (deduce) about the truth value of the WHOLE.
Appendix C starts with knowing the truth of the WHOLE compound statement and tells you what you can figure out (deduce) about the PARTS.

You "**decompose**" the whole statement into its component parts by starting with the last column from App A and "working backwards" to the p and q values. Several of these also remind us of some Rules of Replacement from App B ~ can you find DeM, Impl., and Equiv.?

Also, notice that when you need to consider two options (**either… or…**), the truth tree "**branches**" and you have to follow both as you continue analyzing within a longer argument (see Lesson 24 for some examples). By the way, our trees branch *down*, so they look more like a root system. ☺

Add these **symbols** (used in Unit 3) with the names at first and as shortcuts once the names are learned. These should sound familiar if you've learned the "self-talk" for App A:

~ ~ D	Double Negation Decomp:	the opposite of a false statement is a true statement
• D	Conjunction Decomp:	when p•q is true, both p and q are true
~ • D	Neg. Conjunction Decomp:	when p•q is false, **either** p **or** q is false (or both)
∨ D	Disjunction Decomp:	when pvq is true, **either** p **or** q is true (or both)
~ ∨ D	Neg. Disjunction Decomp:	when pvq is false, both p and q are false
⊃ D	Conditional Decomp:	when p⊃q is true, **either** p is false **or** q is true
~ ⊃ D	Neg. Conditional Decomp:	when p⊃q is false, p is true and q is false
≡ D	Biconditional Decomp:	p≡q is true when p and q are **either** both true **or** both false
~ ≡D	Neg. Biconditional Decomp:	p≡q is false when p and q have diff. truth values (**2 options**)