Modus Ponens M.P.	Modus Tollens M.T.	Hypothetical Syllogism Hyp Syll H.S.
Dilemma Dilem	Conjunctive Simplification Simp	Conjunction Conj
Disjunctive Syllogism Disj Syll D.S.	Disjunctive Addition Disj Add Add	Double Thens D. Thens
Biconditional Exchange Bicond (argument form)	Reductio Ad Absurdum R.A.A.	Double Negation Doub Neg D.N.
De Morgan's Laws DeMorg	Contraposition Contrap	Conditional Relation Cond
Biconditional Exchange Bicond (equivalence form)	Double Ifs D. Ifs	Duplication Dupl

if p then q if q then r --- ∴ if p then r	p ⊃ q q ⊃ r --- ∴ p ⊃ r	if p then q not-q --- ∴ not-p p ⊃ q ~q --- ∴ ~p	if p then q p --- ∴ q p ⊃ q p --- ∴ q
p q --- ∴ p and q	p q --- ∴ p & q	p and q --- ∴ p ∴ q p & q --- ∴ p ∴ q	either p or q if p then r if q then s --- ∴ either r or s p ∨ q p ⊃ r q ⊃ s --- ∴ r ∨ s p ∨ q p ⊃ r q ⊃ r --- ∴ r
if p then both q and r ∴ if p then q ∴ if p then r (reversable) p ⊃ (q & r) --- ∴ p ⊃ q ∴ p ⊃ r p ⊃ q p ⊃ r --- ∴ p ⊃ (q & r)	p --- ∴ either p or q ∴ either q or p p --- ∴ p ∨ q ∴ q ∨ p	either p or q not-p --- ∴ q p ∨ q ~p --- ∴ q
p = not (not-p) p = ~(~p)	if p then q if p then not-q --- ∴ not-p p ⊃ q p ⊃ ~q --- ∴ ~p	p if and only if q --- ∴ if p then q ∴ if q then p p ≡ q --- ∴ p ⊃ q ∴ q ⊃ p p ⊃ q q ⊃ p --- ∴ p ≡ q (reversable)
if p then q = not-p, or else q p ⊃ q = ~p v q	if p then q = if not-q then not-p p ⊃ q = ~q ⊃ ~p	not both p and q = not-p or not-q ~(p & q) = ~p ∨ ~q not either p or q = not-p and not-q ~(p ∨ q) = ~p & ~q
p = both p and p p = p & p p = either p or p p = p ∨ p	if both p and q, then r = if p, then, if q, then r = if p, and if q, then r (p & q) ⊃ r = p ⊃ (q ⊃ r)	p if and only if q = if p then q, and, if q then p p ≡ q = (p ⊃ q) & (q ⊃ p)

Commutation Comm	Association Assoc	Distribution Dist
	Logical Truth (Tautology) Taut	

p and (q or r) = (p and q) or (p and r) p & (q V r) = (p & q) V (p & r) p or (q and r) = (p or q) and (p or r) p V (q & r) = (p V q) & (p V r)	(p and q) and r = p and (q and r) (p & q) & r = p & (q & r) (p or q) or r = p or (q or r) (p V q) V r = p V (q V r)	both p and q = both q and p p & q = q & p either p or q = either q or p p V q = q V p
	p V ~p either p or not-p ~(p & ~p) not both p and not-p p ⊃ p if p, then p (p & q) ⊃ p if p and q, then p (p & q) ⊃ q if p and q, then q p ⊃ (p V q) if p, then p or q p ⊃ (q V p) if p, then q or p	