Theorems

Equivalence and true

(3.1)	Axiom, Associativity of $\equiv$	$((p \equiv q) \equiv r) \equiv (p \equiv (q \equiv r))$
(3.2)	Axiom, Symmetry of $\equiv$	$p \equiv q \equiv q \equiv p$
(3.3)	Axiom, Identity of $\equiv$	$true \equiv q \equiv q$
(3.4)		$true$
(3.5)	Reflexivity of $\equiv$	$p \equiv p$

Negation, inequivalence, and false

(3.8)	Axiom, Definition of *false*	$false \equiv \neg true$
(3.9)	Axiom, Distributivity of $\neg$ over $\equiv$	$\neg(p \equiv q) \equiv \neg p \equiv q$
(3.10)	Axiom, Definition of $\not\equiv$	$(p \not\equiv q) \equiv \neg(p \equiv q)$
(3.11)		$\neg p \equiv q \equiv p \equiv \neg q$
(3.12)	Double negation	$\neg\neg p \equiv p$
(3.13)	Negation of *false*	$\neg false \equiv true$
(3.14)		$(p \not\equiv q) \equiv \neg p \equiv q$
(3.15)		$(\neg p \equiv p \equiv false)$
(3.16)	Symmetry of $\not\equiv$	$(p \not\equiv q) \equiv (q \not\equiv p)$
(3.17)	Associativity of $\not\equiv$	$((p \not\equiv q) \not\equiv r) \equiv (p \not\equiv (q \not\equiv r))$
(3.18)	Mutual Associativity	$((p \not\equiv q) \equiv r) \equiv (p \not\equiv (q \equiv r))$
(3.19)	Mutual interchangeability	$p \not\equiv q \equiv r \equiv p \equiv q \not\equiv r$

Disjunction

(3.24)	Axiom, Symmetry of $\vee$	$p \vee q \equiv q \vee p$
(3.25)	Axiom, Associativity of $\vee$	$(p \vee q) \vee r \equiv p \vee (q \vee r)$
(3.26)	Axiom, Idempotency of $\vee$	$p \vee p \equiv p$
(3.27)	Axiom, Distributivity of $\vee$ over $\equiv$	$p \vee (q \equiv r) \equiv p \vee q \equiv p \vee r$
(3.28)	Axiom, Excluded Middle	$p \vee \neg p$
(3.29)	Zero of $\vee$	$p \vee true \equiv true$
(3.30)	Identity of $\vee$	$p \vee false \equiv p$
(3.31)	Distributivity of $\vee$ over $\vee$	$p \vee (q \vee r) \equiv (p \vee q) \vee (p \vee r)$
(3.32)		$p \vee q \equiv p \vee \neg q \equiv p$

Conjunction

(3.35)	Axiom, Golden rule	$p \wedge q \equiv p \equiv q \equiv p \vee q$
(3.36)	Symmetry of $\wedge$	$p \wedge q \equiv q \wedge p$
(3.37)	Associativity of $\wedge$	$(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$
(3.38)	Idempotency of $\wedge$	$p \wedge p \equiv p$
(3.39)	Identity of $\wedge$	$p \wedge true \equiv p$
(3.40)	Zero of $\wedge$	$p \wedge false \equiv false$
(3.41)	Distributivity of $\wedge$ over $\wedge$	$p \wedge (q \wedge r) \equiv (p \wedge q) \wedge (p \wedge r)$
(3.42)	Contradiction	$p \wedge \neg p \equiv false$
(3.43)	Absorption	(a) $p \wedge (p \vee q) \equiv p$
		(b) $p \vee (p \wedge q) \equiv p$
(3.44)	Absorption	(a) $p \wedge (\neg p \vee q) \equiv p \wedge q$
		(b) $p \vee (\neg p \wedge q) \equiv p \vee q$
(3.45)	Distributivity of $\vee$ over $\wedge$	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
(3.46)	Distributivity of $\wedge$ over $\vee$	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
(3.47)	De Morgan	(a) $\neg(p \wedge q) \equiv \neg p \vee \neg q)$
		(b) $\neg(p \vee q) \equiv \neg p \wedge \neg q)$
(3.48)		$p \wedge q \equiv p \wedge \neg q \equiv \neg p$
(3.49)		$p \wedge (q \equiv r) \equiv p \wedge q \equiv p \wedge r \equiv p$
(3.50)		$p \wedge (q \equiv p) \equiv p \wedge q$
(3.51)	Replacement	$(p \equiv q) \wedge (r \equiv p) \equiv (p \equiv q) \vee (r \equiv q)$
(3.52)	Definition of $\equiv$	$p \equiv q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$
(3.53)	Exclusive or	$p \not\equiv q \equiv (\neg p \wedge q) \vee (p \wedge \neg q)$
(3.55)		$(p \wedge q) \wedge r \equiv p \equiv q \equiv r \equiv p \vee q \equiv q \vee r \equiv r \vee p \equiv p \vee q \vee r$

Implication

(3.57)	Axiom, Definition of Implication	$p \Rightarrow q \equiv p \vee q \equiv q$
(3.58)	Axiom, Consequence	$p \Leftarrow q \equiv q \Rightarrow p$
(3.59)	Definition of implication	$p \Rightarrow q \equiv \neg p \vee q$
(3.60)	Definition of implication	$p \Rightarrow q \equiv p \wedge q \equiv p$
(3.61)	Contrapositive	$p \Rightarrow q \equiv \neg q \Rightarrow \neg p$
(3.62)		$p \Rightarrow (q \equiv r) \equiv p \wedge q \equiv p \wedge r$
(3.63)	Distributivity of $\Rightarrow$ over $\equiv$	$p \Rightarrow (q \equiv r) \equiv p \Rightarrow q \equiv p \Rightarrow r$
(3.64)		$p \Rightarrow (q \Rightarrow r) \equiv (p \Rightarrow q) \Rightarrow (p \Rightarrow r)$
(3.65)	Shunting	$p \wedge q \Rightarrow r \equiv p \Rightarrow (q \Rightarrow r)$
(3.66)		$p \wedge (p \Rightarrow q) \equiv p \wedge q$
(3.67)		$p \wedge (q \Rightarrow p) \equiv p$
(3.68)		$p \vee (p \Rightarrow q) \equiv true$
(3.69)		$p \vee (q \Rightarrow p) \equiv q \Rightarrow p$
(3.70)		$p \vee q \Rightarrow p \wedge q \equiv p \equiv q$
(3.71)	Reflexivity of $\Rightarrow$	$p \Rightarrow p \equiv true$
(3.72)	Right zero of $\Rightarrow$	$p \Rightarrow true \equiv true$
(3.73)	Left identity of $\Rightarrow$	$true \Rightarrow p \equiv p$
(3.74)		$p \Rightarrow false \equiv \neg p$
(3.75)		$false \Rightarrow p \equiv true$
(3.76)	Weakening/strengthening	(a) $p \Rightarrow p \vee q$
		(b) $p \wedge q \Rightarrow p$
		(c) $p \wedge q \Rightarrow p \vee q$
		(d) $p \vee (q \wedge r) \Rightarrow p \vee q$
		(e) $p \wedge q \Rightarrow p \wedge (q \vee r)$
(3.77)	Modus ponens	$p \wedge (p \Rightarrow q) \Rightarrow q$
(3.78)		$(p \Rightarrow r) \wedge (q \Rightarrow r) \equiv (p \vee q \Rightarrow r)$
(3.79)		$(p \Rightarrow r) \wedge (\neg p \Rightarrow r) \equiv r$
(3.80)	Mutual implication	$(p \Rightarrow q) \wedge (q \Rightarrow p) \equiv (p \equiv q)$
(3.81)	Antisymmetry	$(p \Rightarrow q) \wedge (q \Rightarrow p) \Rightarrow (p \equiv q)$
(3.82)	Transitivity	(a) $(p \Rightarrow q) \wedge (q \Rightarrow r) \Rightarrow (p \Rightarrow r)$
		(b) $(p \equiv q) \wedge (q \Rightarrow r) \Rightarrow (p \Rightarrow r)$
		(c) $(p \Rightarrow q) \wedge (q \equiv r) \Rightarrow (p \Rightarrow r)$

Leibniz as an axiom

(3.83)	Axiom, Leibniz	$e = f \Rightarrow E^z_e = E^z_f$
(3.84)	Substitution	(a) $(e = f) \wedge E^z_e \equiv (e = f) \wedge E^z_f$
		(b) $(e = f) \Rightarrow E^z_e \equiv (e = f) \Rightarrow E^z_f$
		(c) $q \wedge (e = f) \Rightarrow E^z_e \equiv q \wedge (e = f) \Rightarrow E^z_f$
(3.85)	Replace by *true*	(a) $p \Rightarrow E^z_b \equiv p \Rightarrow E^z_{true}$
		(b) $q \wedge p \Rightarrow E^z_p \equiv q \wedge p \Rightarrow E^z_{true}$
(3.86)	Replace by *false*	(a) $E^z_p \Rightarrow p \equiv E^z_{false} \Rightarrow p$
		(b) $E^z_p \Rightarrow p \vee q \equiv E^z_{false} \Rightarrow p \vee q$
(3.87)	Replace by *true*	$p \wedge E^z_p \equiv p \wedge E^z_{true}$
(3.88)	Replace by *false*	$p \vee E^z_p \equiv p \vee E^z_{false}$
(3.89)	Shannon	$E^z_p \equiv (p \wedge E^z_{true}) \vee (\neg p \wedge E^z_{false})$
(4.1)		$p \Rightarrow (q \Rightarrow p)$
(4.2)	Monotonicity of $\vee$	$(p \Rightarrow q) \Rightarrow (p \vee r \Rightarrow q \vee r)$
(4.3)	Monotonicity of $\wedge$	$(p \Rightarrow q) \Rightarrow (p \wedge r \Rightarrow q \wedge r)$

Proof techniques

(4.4)	Deduction	To prove $P \Rightarrow Q$, assume P and prove Q.
(4.5)	Case analysis	If E^z_{true}, E^z_{false} are theorems, then so is E^z_P.
(4.6)	Case analysis	$(p \vee q \vee r) \wedge (p \Rightarrow s) \wedge (q \Rightarrow s) \wedge (r \Rightarrow s) \Rightarrow s$
(4.7)	Mutual implication	To prove $P \equiv Q$, prove $P \Rightarrow Q$ and $Q \Rightarrow P$.
(4.9)	Proof by contradiction	To prove P, prove $\neg P \Rightarrow false$.
(4.12)	Proof by contrapositive	To prove $P \Rightarrow Q$, prove $\neg Q \Rightarrow \neg P$

General Laws of Quantification

For symmetric and associative binary operator $\star$ with identity u.

(8.13)	**Axiom, Empty range:** $(\star x \mid false : P) = u$
(8.14)	**Axiom, One-point rule:** Provided $\neg occurs(`x', `E')$, $(\star x \mid x = E : P) = P[x := E]$
(8.15)	**Axiom, Distributivity:** Provided each quantification is defined, $(\star x \mid R : P) \star (\star x \mid R : Q) = (\star x \mid R : P \star Q)$
(8.16)	**Axiom, Range split:** Provided $R \wedge S \equiv false$ and each quantification is defined, $(\star x \mid R \vee S : P) = (\star x \mid R \wedge S : P) = (\star x \mid R : P) \star (\star X \mid S : P)$
(8.17)	**Axiom, Range split:** Provided each quantification is defined, $(\star x \mid R \vee S : P) \star (\star x \mid R \wedge S : P) = (\star x \mid R : P) \star (\star x \mid S : P)$
(8.18)	**Axiom, Range split for idempotent $\star$:** Prov. each quant. is defined, $(\star x \mid R \vee S : P) = (\star x \mid R : P) \star (\star x \mid S : P)$
(8.19)	**Axiom, Interchange of dummies:** Provided each quantification is defined, $\neg occurs(`y', `R')$, and $\neg occurs(`x', `Q')$, $(\star x \mid R : (\star y \mid Q : P)) = (\star y \mid Q : (\star x \mid R : P))$
(8.20)	**Axiom, Nesting:** Provided $\neg occurs(`y', `R')$, $(\star x,\, y \mid R \wedge Q : P) = (\star x \mid R : (\star y \mid Q : P))$
(8.21)	**Axiom, Dummy renaming:** Provided $\neg occurs(`y', `R, P')$, $(\star x \mid R : P) = (\star y \mid R[x := y] : P[x := y])$
(8.22)	**Change of dummy:** Provided $\neg occurs(`y', `R, P')$, and f has an inverse, $(\star x \mid R : P) = (\star y \mid R[x := f.y] : P[x := f.y])$
(8.23)	**Split off term:** $(\star i \mid 0 \le i < n+1 : P) = (\star i \mid 0 \le i < n : P) \star P^i_n$

Theorems of the Predicate Calculus

Universal quantification

(9.2)	**Axiom, Trading:**	$(\forall x \mid R : P) \equiv (\forall x \mid : R \Rightarrow P)$
(9.3)	**Trading:**	(a) $(\forall x \mid R : P) \equiv (\forall x \mid : \neg R \vee P)$
		(b) $(\forall x \mid R : P) \equiv (\forall x \mid : R \wedge P \equiv R)$
		(c) $(\forall x \mid R : P) \equiv (\forall x \mid : R \vee P \equiv P)$
(9.4)	**Trading:**	(a) $(\forall x \mid Q \wedge R : P) \equiv (\forall x \mid Q : R \Rightarrow P)$
		(b) $(\forall x \mid Q \wedge R : P) \equiv (\forall x \mid Q : \neg R \vee P)$
		(c) $(\forall x \mid Q \wedge R : P) \equiv (\forall x \mid Q : R \wedge P \equiv R)$
		(d) $(\forall x \mid Q \wedge R : P) \equiv (\forall x \mid Q : R \vee P \equiv P)$
(9.5)	**Axiom, Distributivity of $\vee$ over $\forall$:**	Prov. $\neg occurs(`x', `P')$, $P \vee (\forall x \mid R : Q) \equiv (\forall x \mid R : P \vee Q)$
(9.6)		Provided $\neg occurs(`x', `P')$, $(\forall x \mid R : P) \equiv P \vee (\forall x \mid : \neg R)$
(9.7)	**Distributivity of $\wedge$ over $\forall$:**	Provided $\neg occurs(`x', `P')$, $\neg(\forall x \mid : \neg R) \Rightarrow ((\forall x \mid R : P \wedge Q) \equiv P \wedge (\forall x \mid R : Q))$
(9.8)		$(\forall x \mid R : true) \equiv true$
(9.9)		$(\forall x \mid R : P \equiv Q) \Rightarrow ((\forall x \mid R : P) \equiv (\forall x \mid R : Q))$
(9.10)	**Range weakening/strengthening:**	$(\forall x \mid Q \vee R : P) \Rightarrow (\forall x \mid Q : P)$
(9.11)	**Body weakening/strengthening:**	$(\forall x \mid R : P \wedge Q) \Rightarrow (\forall x \mid R : P)$
(9.12)	**Monotonicity of $\forall$:**	$(\forall x \mid R : Q \Rightarrow P) \Rightarrow ((\forall x \mid R : Q) \Rightarrow (\forall x \mid R : P))$
(9.13)	**Instantiation:**	$(\forall x \mid : P) \Rightarrow P[x := e]$
(9.16)		P is a theorem iff $(\forall x \mid : P)$ is a theorem.

Existential quantification

(9.17)	**Axiom, Generalized De Morgan:**	$(\exists x \mid R : P) \equiv \neg(\forall x \mid R : \neg P)$
(9.18)	**Generalized De Morgan:**	(a) $\neg(\exists x \mid R : \neg P) \equiv (\forall x \mid R : P)$
		(b) $\neg(\exists x \mid R : P) \equiv (\forall x \mid R : \neg P)$
		(c) $(\exists x \mid R : \neg P) \equiv \neg(\forall x \mid R : P)$
(9.19)	**Trading:**	$(\exists x \mid R : P) \equiv (\exists x \mid : R \wedge P)$
(9.20)	**Trading:**	$(\exists \mid Q \wedge R : P) \equiv (\exists x \mid Q : R \wedge P)$
(9.21)	**Distributivity of $\wedge$ over $\exists$:**	Provided $\neg occurs(`x', `P')$, $P \wedge (\exists x \mid R : Q) \equiv (\exists x \mid R : P \wedge Q)$
(9.22)		$(\exists x \mid R : false) \equiv false$
(9.23)	**Distributivity of $\vee$ over $\exists$:**	Provided $\neg occurs(`x', `P')$, $(\equiv x \mid : R) \Rightarrow ((\equiv x \mid R : P \vee Q) \equiv P \vee (\exists x \mid R : Q))$
(9.24)		$(\exists x \mid R : false) \equiv false$
(9.25)	**Range weakening/strengthening:**	$(\exists x \mid R : P) \Rightarrow (\exists x \mid Q \vee R : P)$
(9.26)	**Body weakening/strengthening:**	$(\exists x \mid R : P) \Rightarrow (\exists x \mid R : P \vee Q)$
(9.27)	**Monotonicity of $\exists$:**	$(\forall x \mid R : Q \Rightarrow P) \Rightarrow ((\exists x \mid R : Q) \Rightarrow (\exists x \mid R : P))$
(9.28)	**$\exists$-Introduction:**	$P[x := E] \Rightarrow (\exists x \mid : P)$
(9.29)	**Interchange of quantifications:**	Provided $\neg occurs(`y', `R')$ and $\neg occurs(`x', `Q')$, $(\exists x \mid R : (\forall y \mid Q : P)) \Rightarrow (\forall y \mid Q : (\exists x \mid R : P))$
(9.30)		Provided $\neg occurs(`\hat{x}', `Q')$, $(\exists x \mid R : P) \Rightarrow Q$ is a theorem iff $(R \wedge P)[x := \hat{x}] \Rightarrow Q$ is a theorem

LN1

Inference rule: $\dfrac{P_1, \ldots, P_k}{C}$, where P_i - premises or hypoth., C is concl.

Inference rule asserts that if the premises are theorems, then the conclusion is a theorem.

Inference rule Substitution: E-expression, v - list of variables, F - list of expressions. $\dfrac{E}{E[v := F]}$.

Laws: Reflexivity. $x = x$, Symmetry. $(x = y) = (y = x)$, Transitivity $\dfrac{X = Y, Y = Z}{X = Z}$, Leibnitz $\dfrac{X = Y}{E[z := X] = E[z := Y]}$

A precondition of a statement is an assertion about the program variables in a state in which the statement may be executed. A postcondition is an assertion about the states in which it may terminate.

Hoare Triple - a notation: $\{P\}S\{Q\}$ Assignment $:= \{R[x := E]\}x := E\{R\}$.

LN2

The *dual* P_D of a boolean expression P is constructed by swapping: $true$ and $false$, $\wedge$ and $\vee$, $\equiv$ and $\not\equiv$, $\Rightarrow$ and $\not\Leftarrow$, $\Leftarrow$ and $\not\Rightarrow$.

and, but	becomes	$\wedge$
or	becomes	$\vee$
not	becomes	$\neg$
it is not the case that	becomes	$\neg$
if p then q	becomes	$p \Rightarrow q$
means	becomes	$\equiv$
however	becomes	$\wedge$
;	becomes	$\wedge$

$\mid$	id	$\neg$
F	F	T
T	T	F

		$\wedge$				$\not\equiv$	$\vee$	nor	$\equiv$		$\Leftarrow$		$\Rightarrow$	nand	
F	F	F	F	F	F	F	F	F	T	T	T	T	T	T	T
F	T	F	F	F	F	T	T	F	F	F	F	T	T	T	T
T	F	F	F	T	T	F	T	F	F	T	T	F	F	T	T
T	T	F	T	F	T	F	T	F	T	F	T	F	T	F	T

Definitions

Expression is **satisfied** in state s iff evaluates to *true* in state s.
Expression is **valid** iff it is satisfied in every state.
Valid expression is called a **tautology**
Expression is **satisfiable** iff there is a state in which it is satisfied.
Expression is a **contradiction** iff it evaluates to *false* in every state.
Two expressions are **logically equivalent** iff they evaluate to same truth value in every state.

LN3

Propositional Calculus = Axioms + Inference Rules, Inference Rules: $\dfrac{P}{P[r := Q]}$.

A theorem of our propositional calculus is either 1 an axiom, 2 the conclusion of an inference rule whose premises are theorems, or 3 a boolean expression that, using the inference rules, is proved equal to an axiom or a previously proved theorem. Heuristic: Identify applicable theorems by matching the structure of expressions or sub-expressions. The operators that appear in a boolean expression and the shape of its sub-expressions can focus the choice of theorems to be used in manipulating it. Principle: Structure proofs to avoid repeating the same sub-expression on many lines.

LN4

See: Proof techniques

LN5

A formal logical system, or logic, is a set of rules defined in terms of a set of symbols, a set of formulas constructed from the symbols, a set of distinguished formulas called axioms, and a set of inference rules. The set of formulas is called the language of the logic. The language is defined syntactically; there is no notion of meaning or semantics in a logic per se. A formula is a theorem of the logic, if it is one of the following: an axiom, can be generated from the axioms and already proved theorems using the inference rules. A proof that a formula is a theorem is an argument that shows how the inference rules are used to generate the formula. A logic is consistent if at least one of its formulas is a theorem and at least one is not; otherwise, the logic is inconsistent. Models: We give the formulas a meaning with respect to this domain, 1 by defining which formulas are true statements about the domain, 2 by defining which formulas are false statements about the domain. An interpretation assigns meaning to the: operators of a logic, constants of a logic and variables of a logic. Standard interpretation of expressions of (a) propositional logic For an expression P without variables, let eval(P) be the value of P. Let Q be any expression, and let s be a state that gives values to all the variables of Q. Define Q(s) to be a copy of Q in which all its variables are replaced by their corresponding values in state s. Then function f given by f (Q) = eval(Q(s)) is an interpretation for Q.

Definitions

Let S be a set of interpretations for a logic and F be a formula of the logic. F is **satisfiable** (under S) iff at least one interpretation of S maps F to true. F is **valid** (under S) iff every interpretation in S maps F to true. An **interpretation** is a model for a logic iff every theorem is mapped to true by the interpretation. A logic is **sound** iff every theorem is valid. A logic is **complete** iff every valid formula is a theorem. **Soundness** means that the theorems are true statements about the domain of discourse, **Completeness** means that every valid formula can be proved. A **sound and complete logic** allows exactly the valid formulas to be proved. A boolean expression is **satisfied** in state s iff it evaluates to true in state s. A boolean expression is **valid** iff it is satisfied in every state. A valid boolean expression is called a **tautology**. A boolean expression is **satisfiable** iff there is a state in which it is satisfied. The atomic proposition is a type of statement, which contains a truth value that can be true or false.

Peano Arithmetic

Symbols: S, o, +, ·, <, = Formulas: φ
Axioms: The axioms of PA are: (1) $\forall x(Sx \neq 0)$
(2) $\forall x, y((Sx = Sy) \rightarrow x = y)$
(3) $(\varphi[0] \wedge \forall x(\varphi[x] \rightarrow \varphi[Sx])) \rightarrow \forall x(\varphi[x])$, for any formula φ in PA.
(4) $\forall x(x + 0) = x$ (5) $\forall x, y(x + Sy = S(x + y))$ (6) $\forall(x \cdot 0 = 0)$
(7) $\forall x, y(x \cdot Sy = (x \cdot y) + x)$
Natural Deduction is a version of Propositional Logic often better suited for formal proofs. Logicians express this relationship between a theorem and the formulas assumed for its proof as the sequent: $A_0, \ldots, A_n \vdash Q$ or $\vdash Q$, where L is the name of the logic with axioms $A_0, \ldots, A_n$. Symbol $\vdash$ is called the "turnstile", and the A_i are called the premises of the sequent. The sequent $A_0, \ldots, A_0 \vdash$ Q is read as Q is provable from $A_0, \ldots, A_n$ (The order of the A_i is immaterial.) The sequent $\vdash_L$ Q is read as "Q is provable in logic L" - i.e. using the axioms of L.

Constructive Propositional Logic

(1) A proof of $p \wedge q$ is given by presenting a proof of p and a proof of q (2) A proof of $p \vee q$ is given by presenting either a proof of p or a proof of q (3) A proof of $p \rightarrow q$ is a procedure that permits us to transform a proof of p into a proof of q. (4) The constant false, which is a contradiction, has no proof. (5) A proof of $\neg p$ is a procedure that transforms any hypothetical proof of p into a proof of a contradiction ($p \vdash false$ i.e., false is provable from p).
Rules for Constructive Natural Deduction:
Introduction rules:
———
$\wedge - I$: $\dfrac{\vdash P, \vdash Q}{\vdash P \wedge Q}$ $\vee - I_l$: $\dfrac{\vdash P}{\vdash P \vee Q}$ $\vee - I_r$: $\dfrac{\vdash Q}{\vdash P \vee Q}$
$\rightarrow - I$: $\dfrac{P_1, \ldots, P_n \vdash Q}{\vdash P_1 \wedge \ldots \wedge P_n \rightarrow Q}$ $false - I$: (none)
———
Elimination Rules:
———
$\wedge - E \dfrac{P \wedge Q}{P}, \dfrac{P \wedge Q}{Q}$ $\vee - E \dfrac{P \wedge Q, P \rightarrow R, Q \rightarrow R}{R}$ $\rightarrow - E \dfrac{P, P \rightarrow Q}{Q}$
$\equiv - E_l \dfrac{\vdash P \wedge Q}{\vdash P}, \wedge - E_r \dfrac{\vdash P \wedge Q}{Q}$ $\vee - E \dfrac{\vdash P \vee Q, P \vdash R, Q \vdash R}{\vdash R}$
$\rightarrow - E : \dfrac{\vdash P, \vdash P] \rightarrow Q}{\vdash Q} F - E \dfrac{\vdash F}{\vdash P}$
$P \equiv Q$ denotes $(P \rightarrow Q) \wedge (Q \rightarrow P)$ ——— $\neg P$ denotes $P \rightarrow F$ T denotes $\neg F$
$\neg\neg p \rightarrow p$ is NOT a theorem $p \rightarrow \neg\neg p$ is a theorem
$p \vee \neg p$ is NOT a theorem $\neg\neg(p \vee \neg p)$ is a theorem.
Theorem-Soundness: An inference rule is considered sound if a formula derived using it is valid whenever the premises used in the inference are theorems.
Model-Soundness: An inference rule is considered sound if a formula derived using it is valid whenever the premises used in the inference are valid.

LN6

In a textual substitution $E[x := F]$, x and F must have the same type. a notion of subtypes: for example, the natural numbers **N** are a subset of the integers **Z**, so l : **Z** and l : **N** are both suitable declarations. a notion overloading: we need a notion of subtypes, as well as a notion of overloading of both constants and operators, so that the same constants and operators can be used in more than one way. a notion of polymorphism: we also need a notion of polymorphism; as an example function =: t × t → **B** is polymorphic because it is defined for any type t.
$\Sigma_{i=1}^{n} e$ is any expression. $\Sigma(i|1 \leq i < n : e)$ Linear notation
Let $*$ be any binary operator that satisfy:
Sym/Comm: $b * c = c * b$ Assoc.: $(b * c) * d = b * (c * d)$
Id. u: $u * b = b = b * u$ A set of values together with an operator $*$ that satisfy the above is called an Abelian monoid.
The general form of a quantification over $*$ is exemplified by
$*(x : t1, y : t2|R : P)$ Variables x and y are distinct. They are called the bound variables or dummies of the quantification. t1 and t2 are the types of dummies x and y If t1 and t2 are the same type, we may write $*(x, y : t1|R : P)$ R, a boolean expression, is the range of the quantification R may refer to dummies x and y. If the range is omitted, as in $*(x, y : t1| : P)$, then the range true is meant. P, an expression, is the body of the quantification. P may refer to dummies x and y. Expression $*(x : X|R : P)$ denotes the application of operator $*$ to the values P for all x in X for which range R is true.
Free and Bound occurrences in a variable:
The occurrence of i in the expression i is free. Suppose an occurrence of i in expression E is free. Then that same occurrence of i is free in (E), in function application $f(\ldots, E, \ldots)$, and in $*(x|E : F)$ and $*(x|F : E)$ provided i is not one of the dummies in list x.
Let an occurrence of i be free in an expression E. That occurrence of i is bound (to dummy i) in the expression $*x|E : F)$ and $*(x|F : E)$ if i is one of the dummies in list x. Suppose an occurrence of i is bound in expression E. Then it is also bound (to the same dummy) in (E), in function application f (..., E,...) , and in $*(x|E : F)$ and $*(x|F : E)$.

Textual Substitution

Provided $\neg occurs('y', 'x, F')$, i.e. a dummy of list y will have to be replaced by a fresh variable if that dummy occurs free in x or F.
$*(y|R : P)[x := F] = *(y|R[x := F] : P[x := F])$
Assume that the operator $*$ is symmetric and associative and has an identity u.
Two additional inferences rules allow substitution of equals for equals in the range and body of a quantification (Leibniz).

$$\frac{P = Q}{*(x|E[z := P] : S) = *(x|E[z := Q] : S)}$$

$$\frac{R \rightarrow P = Q}{*(x|R : E[z := P]) = *(x|R : E[z := Q])}$$

Operation $*$ is idempotent iff $x * x = x$ for all x. Quantifiers '$\vee$', '$\wedge$', '$\cup$', '$\cap$' are idempotent, while '+' and '·' are not.

LN7

A predicate-calculus formula is a boolean expression in which some boolean variables may have been replaced by: Predicates : applications of boolean functions whose arguments may be of types other than **B**, Universal and existential quantification.

LN7a

$R_i(x_1, \ldots, x_n)$ - atomic formula, n is and arity of the relational symbol R_i. All appearances of R_i must have the same arity.
Φ is a formula iff: Φ is atomic. $\Phi = \Phi_1 \wedge \Phi_2, \Phi = \Phi_1 \vee \Phi_2, \Phi = \Phi\neg\Phi$ where Φ_1 and Φ_2 are formulas. $\Phi = \exists[\Phi], \Phi = \forall[\Phi]$
Prenex Normal Form: All quantifiers appear in the front of the formula. We assume all our formulas are in prenex normal form. It can be proved that every formula has its equivalent prenex normal form. Free variable: not bound by any quantifier Sentence, statement: no free variables.
A model (interpretation, structure) is a tuple $M = (U, P_1, \ldots, P_k)$, where U is a universe over which the variables may take values, P_i is a relation assigned to the symbol R_i. A language of a model is the set of all formulas of the model. If the formula Φ is true in a model M, we say that M is a Model of Φ.
$\Phi = \forall x.\forall y.R_1(x, y) \vee R_1(y, x)$ Model $M_1 : U$ - natural numbers, P1 is $\leq$ (we write $a \leq b$ instead of $\leq (a, b)$ or $P_1(a, b))\Phi M_1 = \forall x.\forall y.x \leq y \vee y \leq x$ - the formula ΦM_1 is true so M_1 is a model of Φ. Model M_2: U - natural numbers, P_1 is $<$ (we write $a < b$ instead of $< (a, b)$ or $P_1(a, b))$ $\Phi M_2 = \forall x.\forall y.x < y \vee y < x$, the formula ΦM_2 is false so M_2 is not a model of Φ.
Examples of English to Predicate logic:
(a) The natural number 1 is the only natural number that is smaller than positive integer p and divides p. $(\forall d|1 < d < p : \neg(\exists v|0 \leq v : d \cdot v = p))$ (c) Adding two odd integers yields an even number. (Use only addition and multiplication; do not use division, mod, or predicates even.x and odd.x)
$(\forall x, y : \mathbf{Z}|(\exists i, j : \mathbf{Z}| : x = 2 \cdot i + 1 \wedge y = 2 \cdot j + 1) : (\exists k : \mathbf{Z}| : x + y = 2 \cdot k))$ $\forall$ Everybody, $\exists$ Somebody, $\neg\exists$ Nobody.
(a) Everybody loves everybody. $(\forall x : P| : (\forall y : P| : loves(x, y)))$ (b) Nobody loves everybody. $\neg(\exists x : P| : (\forall y : P| : loves(x, y)))$ (c) Somebody loves nobody. $(\exists x : P| : \neg(\exists y : P| : loves(x, y)))$