Tables

Part I

Truth-Tables						
P	Q	$P \wedge Q$	$P \vee Q$	$\neg P$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
0	0	0	0	1	1	1
0	1	0	1		1	0
1	0	0	1	0	0	0
1	1	1	1		1	1

Equivalences for connectives

Commutativity: $P \wedge Q \stackrel{val}{==} Q \wedge P$, $P \vee Q \stackrel{val}{==} Q \vee P$, $P \Leftrightarrow Q \stackrel{val}{==} Q \Leftrightarrow P$	**Associativity:** $(P \wedge Q) \wedge R \stackrel{val}{==} P \wedge (Q \wedge R)$, $(P \vee Q) \vee R \stackrel{val}{==} P \vee (Q \vee R)$, $(P \Leftrightarrow Q) \Leftrightarrow R \stackrel{val}{==}$ $P \Leftrightarrow (Q \Leftrightarrow R)$
Idempotence: $P \wedge P \stackrel{val}{==} P$, $P \vee P \stackrel{val}{==} P$	**Double Negation:** $\neg\neg P \stackrel{val}{==} P$
Inversion: $\neg\texttt{True} \stackrel{val}{==} \texttt{False}$, $\neg\texttt{False} \stackrel{val}{==} \texttt{True}$	**True/False-elimination:** $P \wedge \texttt{True} \stackrel{val}{==} P$, $P \wedge \texttt{False} \stackrel{val}{==} \texttt{False}$, $P \vee \texttt{True} \stackrel{val}{==} \texttt{True}$, $P \vee \texttt{False} \stackrel{val}{==} P$
Negation: $\neg P \stackrel{val}{==} P \Rightarrow \texttt{False}$	**Contradiction:** $P \wedge \neg P \stackrel{val}{==} \texttt{False}$ **Excluded Middle:** $P \vee \neg P \stackrel{val}{==} \texttt{True}$
Distributivity: $P \wedge (Q \vee R) \stackrel{val}{==} (P \wedge Q) \vee (P \wedge R)$, $P \vee (Q \wedge R) \stackrel{val}{==} (P \vee Q) \wedge (P \vee R)$	**De Morgan:** $\neg(P \wedge Q) \stackrel{val}{==} \neg P \vee \neg Q$, $\neg(P \vee Q) \stackrel{val}{==} \neg P \wedge \neg Q$
Implication: $P \Rightarrow Q \stackrel{val}{==} \neg P \vee Q$	**Contraposition:** $P \Rightarrow Q \stackrel{val}{==} \neg Q \Rightarrow \neg P$
Bi-implication: $P \Leftrightarrow Q \stackrel{val}{==} (P \Rightarrow Q) \wedge (Q \Rightarrow P)$	**Self-equivalence:** $P \Leftrightarrow P \stackrel{val}{==} \texttt{True}$

Weakening rules

$\wedge$-$\vee$-weakening: $P \wedge Q \overset{val}{\models} P$, $P \overset{val}{\models} P \vee Q$	**Extremes:** $\texttt{False} \overset{val}{\models} P$, $P \overset{val}{\models} \texttt{True}$
Monotonicity: If $P \overset{val}{\models} Q$, then $P \wedge R \overset{val}{\models} Q \wedge R$, If $P \overset{val}{\models} Q$, then $P \vee R \overset{val}{\models} Q \vee R$	

Equivalences for quantifiers

Bound Variable: $\forall_x[P : Q] \overset{val}{==} \forall_y[P[y \text{ for } x] : Q[y \text{ for } x]]$, $\exists_x[P : Q] \overset{val}{==} \exists_y[P[y \text{ for } x] : Q[y \text{ for } x]]$	
Domain Splitting: $\forall_x[P \vee Q : R] \overset{val}{==} \forall_x[P : R] \wedge \forall_x[Q : R]$, $\exists_x[P \vee Q : R] \overset{val}{==} \exists_x[P : R] \vee \exists_x[Q : R]$	
One-element: $\forall_x[x = n : Q] \overset{val}{==} Q[n \text{ for } x]$, $\exists_x[x = n : Q] \overset{val}{==} Q[n \text{ for } x]$	**Empty Domain:** $\forall_x[\texttt{False} : Q] \overset{val}{==} \texttt{True}$, $\exists_x[\texttt{False} : Q] \overset{val}{==} \texttt{False}$
Domain Weakening: $\forall_x[P \wedge Q : R] \overset{val}{==} \forall_x[P : Q \Rightarrow R]$, $\exists_x[P \wedge Q : R] \overset{val}{==} \exists_x[P : Q \wedge R]$	**De Morgan:** $\neg\forall_x[P : Q] \overset{val}{==} \exists_x[P : \neg Q]$, $\neg\exists_x[P : Q] \overset{val}{==} \forall_x[P : \neg Q]$

Part II

<table>
<tr><th colspan="2">Derivation rules for $\wedge$ and $\Rightarrow$</th></tr>
<tr>
<td>$\wedge$-introduction:

$\vdots$
(k) P
$\vdots$
(l) Q
$\vdots$
{ $\wedge$-intro on (k) and (l): }
(m) $P \wedge Q$</td>
<td>$\wedge$-elimination:

||||
(k) $P \wedge Q$
||||
{ $\wedge$-elim on (k): }
(m) P resp. Q</td>
</tr>
<tr>
<td>$\Rightarrow$-introduction:

{ Assume: }
(k) $\boxed{P}$
$\vdots$
$(m-1)$ Q
{ $\Rightarrow$-intro on
(k) and $(m-1)$: }
(m) $P \Rightarrow Q$</td>
<td>$\Rightarrow$-elimination:

||||
(k) $P \Rightarrow Q$
||||
(l) P
||||
{ $\Rightarrow$-elim on
(k) and (l): }
(m) Q</td>
</tr>
</table>

Derivation rules for ¬ , `False` and ¬¬

¬-introduction: { Assume: } (k) | $\boxed{P}$ | ⋮ (m−1) | `False` { ¬-intro on (k) and (m−1): } (m) ¬P	**¬-elimination:** * ⦙ (k) ¬P ⦙ (l) P ⦙ { ¬-elim on (k) and (l): } (m) `False`
`False`-introduction: * ⋮ (k) ¬P ⋮ (l) P ⋮ { `False`-intro on (k) and (l): } (m) `False`	**`False`-elimination:** ⦙ (k) `False` ⦙ { `False`-elim on (k): } (m) P
¬¬-introduction: ⋮ (k) P ⋮ { ¬¬-intro on (k): } (m) ¬¬P	**¬¬-elimination:** ⦙ (k) ¬¬P ⦙ { ¬¬-elim on (k): } (m) P

* Note: ¬-elim and `False`-intro are similar, but differ in use. See Section 14.3, I.

Derivation rules for $\vee$ and $\Leftrightarrow$

$\vee$-introduction:

{ Assume: }
(k) $\boxed{\neg P}$
$\vdots$
$(m-1)$ Q
{ $\vee$-intro on (k) and $(m-1)$: }
(m) $P \vee Q$

- - - - resp. - - - -

{ Assume: }
(k) $\boxed{\neg Q}$
$\vdots$
$(m-1)$ P
{ $\vee$-intro on (k) and $(m-1)$: }
(m) $P \vee Q$

$\vee$-elimination:

|||
(k) $P \vee Q$
|||
(l) $\neg P$
|||
{ $\vee$-elim on (k) and (l): }
(m) Q

- - - - resp. - - - -

|||
(k) $P \vee Q$
|||
(l) $\neg Q$
|||
{ $\vee$-elim on (k) and (l): }
(m) P

$\Leftrightarrow$-introduction:

$\vdots$
(k) $P \Rightarrow Q$
$\vdots$
(l) $Q \Rightarrow P$
$\vdots$
{ $\Leftrightarrow$-intro on (k) and (l): }
(m) $P \Leftrightarrow Q$

$\Leftrightarrow$-elimination:

|||
(k) $P \Leftrightarrow Q$
|||
{ $\Leftrightarrow$-elim on (k): }
(m) $P \Rightarrow Q$ resp. $Q \Rightarrow P$

Proof by contradiction

{ Assume: }

(k) $\neg P$

⋮

(m − 1) `False`

{ ¬-intro on (k) and (m − 1),
followed by ¬¬-elim: }

(m) P

Proof by case distinction

|||

(k) $P \vee Q$

|||

(l) $P \Rightarrow R$

|||

(m) $Q \Rightarrow R$

|||

{ Case distinction on (k), (l) and (m): }

(n) R

Derivation rules for ∀ and ∃	
∀-introduction: { Assume: } (k) $\underline{\text{var}}\ x;\ P(x)$ ⋮ $(m-1)$ $Q(x)$ { ∀-intro on (k) and $(m-1)$: } (m) $\forall_x[P(x) : Q(x)]$	**∀-elimination:** ⫼ (k) $\forall_x[P(x) : Q(x)]$ ⫼ (l) $P(a)$ ⫼ { ∀-elim on (k) and (l): } (m) $Q(a)$ (a must be an object being 'available' in line (l))
∃-introduction: { Assume: } (k) $\forall_x[P(x) : \neg Q(x)]$ ⋮ $(m-1)$ `False` { ∃-intro on (k) and $(m-1)$: } (m) $\exists_x[P(x) : Q(x)]$	**∃-elimination:** ⫼ (k) $\exists_x[P(x) : Q(x)]$ ⫼ (l) $\forall_x[P(x) : \neg Q(x)]$ ⫼ { ∃-~~intro~~ on (elim) (k) and (l): } (m) `False`

Alternative derivation rules for $\exists$

$\exists^*$**-introduction:**	$\exists^*$**-elimination:**
$\vdots$	$\vert\vert\vert$
(k) $P(a)$	(k) $\exists_x[P(x) : Q(x)]$
$\vdots$	$\vert\vert\vert$
(l) $Q(a)$	{ $\exists^*$-elim on (k): }
{ $\exists^*$-intro on (k) and (l): }	(m) Pick an x with $P(x)$ and $Q(x)$.
(m) $\exists_x[P(x) : Q(x)]$	(x in line (m) must be 'new')
(a must be an object being 'available' in lines (k) and (l))	

Part III

Sets	
$A \subseteq B \stackrel{\text{def}}{=} \forall_x[x \in A : x \in B]$	$A = B \stackrel{\text{def}}{=} A \subseteq B \wedge B \subseteq A$
$A \cap B \stackrel{\text{def}}{=} \{x \in \mathbf{U} \mid x \in A \wedge x \in B\}$	$A \cup B \stackrel{\text{def}}{=} \{x \in \mathbf{U} \mid x \in A \vee x \in B\}$
$A^{\mathrm{C}} \stackrel{\text{def}}{=} \{x \in \mathbf{U} \mid \neg(x \in A)\}$	$A \backslash B \stackrel{\text{def}}{=} \{x \in \mathbf{U} \mid x \in A \wedge \neg(x \in B)\}$
$\mathbf{U} = \{\ x \in \mathbf{U} \mid \mathtt{True}\}$	$\emptyset = \{\ x \in \mathbf{U} \mid \mathtt{False}\}$

Property of $\in$: $t \in \{x \in \mathbf{D} \mid P(x)\} \stackrel{val}{==} t \in \mathbf{D} \ \wedge \ P(t)$	
Property of $\subseteq$: $A \subseteq B \ \wedge \ t \in A \ \stackrel{val}{\models} \ t \in B$	**Properties of $=$:** $A = B \stackrel{val}{==} \forall_x[x \in A \Leftrightarrow x \in B]$ $A = B \ \wedge \ t \in A \ \stackrel{val}{\models} \ t \in B$ $A = B \ \wedge \ t \in B \ \stackrel{val}{\models} \ t \in A$
Property of $\cap$: $t \in A \cap B \stackrel{val}{==} t \in A \ \wedge \ t \in B$	**Property of $\cup$:** $t \in A \cup B \stackrel{val}{==} t \in A \ \vee \ t \in B$
Property of $^{\mathrm{C}}$: $t \in A^{\mathrm{C}} \stackrel{val}{==} \neg(t \in A)$	**Property of $\backslash$:** $t \in A \backslash B \stackrel{val}{==} t \in A \ \wedge \ \neg(t \in B)$
Properties of $\mathbf{U}$: $t \in \mathbf{U} \stackrel{val}{==} \mathtt{True}$ $A = \mathbf{U} \stackrel{val}{==} \forall_x[x \in A : \mathtt{True}]$	**Properties of $\emptyset$:** $t \in \emptyset \stackrel{val}{==} \mathtt{False}$ $A = \emptyset \stackrel{val}{==} \forall_x[x \in A : \mathtt{False}]$
Property of $\mathcal{P}$: $C \in \mathcal{P}(A) \stackrel{val}{==} C \subseteq A$	**Properties of $\times$:** $(a, b) \in A \times B \stackrel{val}{==} a \in A \ \wedge \ b \in B$ $(a, b) = (a', b') \stackrel{val}{==} a = a' \wedge b = b'$

Mappings

Property of 'mapping' $F : A \to B$ **:**
$\forall_x[x \in A : \exists^1_y[y \in B : F(x) = y]]$

Image and source

Let $F : A \to B$ be a mapping, $A' \subseteq A$ and $B' \subseteq B$

the *image* of A': $F(A') =$ $\{b \in B \mid \exists_x[x \in A' : F(x) = b]\}$	the *source* of B': $F^{\leftarrow}(B') =$ $\{a \in A \mid F(a) \in B'\}$
Properties of 'image': $x \in A' \overset{val}{\models} F(x) \in F(A')$ $y \in F(A') \overset{val}{==}$ $\exists_x[x \in A' : F(x) = y]$	**Property of 'source':** $x \in F^{\leftarrow}(B') \overset{val}{==} F(x) \in B'$

Special mappings

Property of 'surjection' for $F : A \to B$ **:** $\forall_y[y \in B :$ $\exists_x[x \in A : F(x) = y]]$	**Property of 'injection' for** $F : A \to B$ **:** $\forall_{x_1,x_2}[x_1, x_2 \in A :$ $(F(x_1) = F(x_2)) \Rightarrow (x_1 = x_2)]$

Property of 'bijection' for $F : A \to B$ **:**
$\forall_y[y \in B : \exists^1_x[x \in A : F(x) = y]]$

Property of 'inverse function' $F^{-1} : B \to A$ **for bijection** $F : A \to B$ **:**
$F(x) = y \overset{val}{==} F^{-1}(y) = x$

Property of 'composite mapping' $G \circ F : A \to C$ **for** $F : A \to B$ **and** $G : B \to C$ **:**
$G \circ F(x) = z \overset{val}{==} G(F(x)) = z$

Standard derivation rules for induction

Induction: $\vdots$ (k) $A(0)$ $\vdots$ (l) $\forall_i[i \in \mathbb{N} :$ $A(i) \Rightarrow A(i{+}1)]$ { Induction on (k) and (l): } (m) $\forall_n[n \in \mathbb{N} : A(n)]$	**Induction from $a \in \mathbb{Z}$:** $\vdots$ (k) $A(a)$ $\vdots$ (l) $\forall_i[i \in \mathbb{Z} \wedge i \geq a :$ $A(i) \Rightarrow A(i{+}1)]$ { Induction on (k) and (l): } (m) $\forall_n[n \in \mathbb{Z} \wedge n \geq a : A(n)]$
Strong induction: $\vdots$ (l) $\forall_k[k \in \mathbb{N} :$ $\forall_j[j \in \mathbb{N} \wedge j{<}k : A(j)]$ $\Rightarrow A(k)]$ { Strong induction on (l): } (m) $\forall_n[n \in \mathbb{N} : A(n)]$	**Strong induction from $a \in \mathbb{Z}$:** $\vdots$ (l) $\forall_k[k \in \mathbb{Z} \wedge k \geq a :$ $\forall_j[j \in \mathbb{Z} \wedge a{\leq}j{<}k : A(j)]$ $\Rightarrow A(k)]$ { Strong induction on (l): } (m) $\forall_n[n \in \mathbb{Z} \wedge n \geq a : A(n)]$

Special relations

R on A is *reflexive* if $\forall_x[x \in A : xRx]$

R on A is *irreflexive* if $\forall_x[x \in A : \neg(xRx)]$

R on A is *symmetric* if $\forall_{x,y}[x, y \in A : xRy \Rightarrow yRx]$

R on A is *antisymmetric* if $\forall_{x,y}[x, y \in A : (xRy \wedge yRx) \Rightarrow x = y]$
R on A is *strictly antisymmetric* if $\forall_{x,y}[x, y \in A : \neg(xRy \wedge yRx)]$

R on A is *transitive* if $\forall_{x,y,z}[x, y, z \in A : (xRy \wedge yRz) \Rightarrow xRz]$

R on A is *linear* if $\forall_{x,y}[x, y \in A : xRy \vee yRx \vee x = y]$

Equivalence relation

R on A is an *equivalence relation* if
R is reflexive and symmetric and transitive

Orderings

R on A is a *quasi-ordering* if R is reflexive and transitive

R on A is a *reflexive ordering* if
R is reflexive, antisymmetric and transitive
R on A is an *irreflexive ordering* if
R is irreflexive, strictly antisymmetric and transitive

R on A is a *reflexive linear ordering* if
R is a reflexive ordering which is also linear
R on A is an *irreflexive linear ordering* if
R is an irreflexive ordering which is also linear

Let $\langle A, R_1\rangle$ and $\langle B, R_2\rangle$ be irreflexive orderings.
The corresponding *lexicographic ordering* $\langle A \times B, R_3\rangle$ is defined by:
$(x, y)R_3(x', y')$ if $xR_1x' \vee (x = x' \wedge yR_2y')$

Special relations and orderings: overview

A is a **set**, R is a **relation** on A

R on A is:	equi-valence relation	quasi-ordering	(partial) ordering		linear ordering	
			refl.	irrefl.	refl.	irrefl.
reflexive	$\surd$	$\surd$	$\surd$		$\surd$	
irreflexive				$\surd$		$\surd$
symmetric	$\surd$					
antisymmetric			$\surd$		$\surd$	
str. antisymm.				$\surd$		$\surd$
transitive	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$
linear					$\surd$	$\surd$
examples:	$=$ $\stackrel{val}{==}$	$\Rightarrow$	$\subseteq$ $\models^{val}$	$\subset$	$\leq$, $\geq$	$<$, $>$

Extreme elements
Let $\langle A, R\rangle$ be a **reflexive ordering** and $A' \subseteq A$
$m \in A'$ is a *maximal element* of A' if $\forall_x[x \in A' : mRx \Rightarrow x = m]$ $m \in A'$ is a *minimal element* of A' if $\forall_x[x \in A' : xRm \Rightarrow x = m]$
$m \in A'$ is the *maximum* of A' if $\forall_x[x \in A' : xRm]$ $m \in A'$ is the *minimum* of A' if $\forall_x[x \in A' : mRx]$
Let $\langle A, S\rangle$ be an **irreflexive ordering** and $A' \subseteq A$
$m \in A'$ is a *maximal element* of A' if $\forall_x[x \in A' : \neg(mSx)]$ $m \in A'$ is a *minimal element* of A' if $\forall_x[x \in A' : \neg(xSm)]$
$m \in A'$ is the *maximum* of A' if $\forall_x[x \in A' \wedge x \neq m : xSm]$ $m \in A'$ is the *minimum* of A' if $\forall_x[x \in A' \wedge x \neq m : mSx]$
Upper and lower bounds
Let $\langle A, R\rangle$ be a **reflexive ordering** and $A' \subseteq A$
$b \in A$ is an *upper bound* of A' if $\forall_x[x \in A' : xRb]$ $a \in A$ is a *lower bound* of A' if $\forall_x[x \in A' : aRx]$