1. THE PROPOSITIONAL CALCULUS PL

YIANNIS N. MOSCHOVAKIS

Contents

§1. Syntax of PL. The **symbols** of *propositional logic* (or *the propositional calculus*) are

$$(\;) \; \neg \; \wedge \; \vee \; \rightarrow \; \mathbf{A}_0 \; \mathbf{A}_1, \ldots$$

where $\neg$ is read "not", $\wedge$ is read "and", $\vee$ is read "or" and $\rightarrow$ is read "implies". These are distinct objects and none of them is a sequence of any of the others. We call $\mathbf{A}_0, \mathbf{A}_1, \ldots$ *sentential* or *propositional symbols* or *variables*, and intuitively they stand for unspecified sentences like "it is raining", "there are infinitely many prime numbers", etc. We use the **metavariables**

$$p, q, r, p_1, q_1, \ldots ,$$

to name arbitrary propositional symbols, as we use the variables x, y, z in algebra to name arbitrary numbers. We will also use Greek letters

$$\alpha, \beta, \gamma, \ldots , \phi, \chi, \psi, \ldots$$

(perhaps with subscripts) to vary over *strings* (or expressions), i.e., finite sequences of symbols. We use the symbol "$\equiv$" for the identity relation on strings of symbols and we denote the *concatenation* of two strings by juxtaposition, so that

$$\text{if } \alpha \equiv \mathbf{A}_0) \text{ and } \beta \equiv \neg(\mathbf{A}_{17}, \text{ then } \alpha\beta \equiv \mathbf{A}_0)\neg(\mathbf{A}_{17}$$

1A. Formulas. The **formulas** (or *well formed formulas*) of PL are defined recursively by the following three clauses:

(a) Each propositional variable $\mathbf{A}_i$ is a formula (as a string of length 1).
(b) If ϕ and ψ are formulas, then the strings

$$(\neg\phi) \quad (\phi \land \psi) \quad (\phi \lor \psi) \quad (\phi \to \psi)$$

are also formulas.
(c) No string is a formula except by virtue of (a) or (b).

This is a bit vague, and it is often abbreviated by the still vaguer (but very suggestive) "recursive definition" in which we read " $\mid$ " as *or*:

$$\phi :\equiv \mathbf{A}_i \mid (\neg\phi_1) \mid (\phi_1 \land \phi_2) \mid (\phi_1 \lor \phi_2) \mid (\phi_1 \to \phi_2)$$

The rigorous definition is as follows:

1A.1. **Definition** (Formulas)**.** A set S of strings is *propositionally closed* if it contains all the propositional variables $\mathbf{A}_i$ (as strings of length 1) and is closed under the sentential connectives: i.e., if $\alpha \in S$, then $(\neg\alpha) \in S$, and if α, β are any two strings in S and $\bullet$ is any binary connective, then the string $(\alpha \bullet \beta)$ is also in S.

A string is a formula if it belongs to every propositionally closed set S.

The propositional variables are called **prime** formulas, while the formulas which are not prime are called **composite**. A formula ϕ is a **subformula** of a formula ψ if for suitable strings α, β, $\psi \equiv \alpha\phi\beta$.

The rigorous definition 1A.1 gives us a very useful method to prove that every formula has a certain property P, by showing that *the set of strings which have property P is propositionally closed.* In other words, to prove that every formula has a certain property P, it is enough to check three things:

(1) Every propositional variable $\mathbf{A}_i$ has property P.
(2) If ϕ has property P, then $(\neg\phi)$ has property P.
(3) If ϕ and ψ have property P, then for any of the three binary connectives $\bullet$, $(\phi \bullet \psi)$ has property P.

This method of proof is called (structural) **induction on formulas**. For example:

1A.2. **Lemma.** (1) *Parentheses match in every formula: i.e., the number of left parentheses which occur in a formula ϕ is equal to the number of right parentheses which occur in ϕ.*

(2) *If α is a proper, non-empty initial part of a formula ϕ, then the number of left parentheses in α is greater than the number of right parentheses in α.*

Proof. (1) The set S of all "balanced" strings (in which parentheses match) is (very easily) propositionally closed, and so it contains all formulas.

We leave (2) for Problem x1.3. $\dashv$

1A.3. **Theorem** (Unique readability). *For every formula ϕ, exactly one of the following is true:*

(1) *ϕ is a propositional variable $\mathbf{A}_i$.*
(2) *There is a formula ψ such that $\phi \equiv (\neg\psi)$.*
(3) *There are formulas ψ, χ and a binary connective $\bullet$ such that $\phi \equiv (\psi \bullet \chi)$.*

Moreover, ψ is uniquely determined in Case (2), and $\psi, \chi, \bullet$ are uniquely determined in Case (3).

Proof is by structural induction and we leave it for Problem x1.4*. $\dashv$

This theorem is often called the *Parsing Lemma* for the propositional calculus. Its last assertion implies, in particular, that if

$$(\psi_1 \bullet_1 \chi_1) \equiv (\psi_2 \bullet_2 \chi_2)$$

for any formulas $\psi_1, \chi_1, \psi_2, \chi_2$, then $\psi_1 \equiv \psi_2, \bullet_1 \equiv \bullet_2$, and $\chi_1 \equiv \chi_2$.

1B. Structural recursion. The *main connective* of a formula $(\psi \bullet \chi)$ is $\bullet$ and its *immediate parts* are ϕ and ψ; similarly, the main connective of $(\neg\psi)$ is $\neg$ and its immediate part is ψ. Theorem 1A.3 insures that each composite formulas ϕ has a uniquely determined main connective and uniquely determined immediate parts, which are shorter formulas than ϕ. This means that we can define a function $F(\phi)$ on formulas by specifying outright the value $F(\mathbf{A}_i)$ of F on prime formulas, and then showing how to compute $F(\phi)$ for composite ϕ using the value of F on the parts of ϕ. This sort of definition is justified by induction on the length of formulas, and it is called *structural recursion*.

1C. Misspellings and abbreviations. In practice, we never put down syntactically correct formulas because it is very tedious—too many parentheses; we give instead instructions on how to construct specific formulas, typically by putting down "mispelled" versions of formulas—with **metavariables** $p, q, r, p_1, \dots$ instead of the formal $\mathbf{A}_i$, without all the parentheses or with brackets in place of some of them, etc. For example,

$$p \to (q \wedge p) \text{ may stand for } (\mathsf{A}_2 \to (\mathsf{A}_{27} \wedge \mathsf{A}_2))$$

and

$$(p \to q) \to \Big[(p \to (q \to r)) \to (p \to r)\Big]$$
$$\text{stands for } ((p \to q) \to ((p \to (q \to r)) \to (p \to r)))$$

with specific propositional variables in place of p, q, r.

Similarly, when we write $(\phi \vee \psi)$, we are using **metavariables over formulas** to refer to any disjunction.

Biconditionals. We will consider the biconditional $\leftrightarrow$ as an abbreviation rather than a primitive connective,

$$(\phi \leftrightarrow \psi) :\equiv ((\phi \to \psi) \wedge (\psi \to \phi))$$

We could go further than that and take as primitive only $\neg$ and $\wedge$ with the rest defined as abbreviations:

$$(\phi \to \psi) :\equiv ((\neg\phi) \vee \psi), \quad (\phi \wedge \psi) :\equiv (\neg((\neg\phi) \vee (\neg\psi)))$$

These definitions agree with our intuitive interpretation of the connectives which we now make precise.

§2. Semantics of PL. Let $\mathbb{B} = \{0, 1\}$ be a fixed set with two members, which (intuitively) we understand as *truth values*: 0 stands for *falsity* and 1 for *truth*. In the primary interpretation of PL the formulas define functions on $\mathbb{B}$ (*bit functions*). These are defined by structural recursion, as follows.

2A. Bit functions and functional completeness. With each formula ϕ and each list $\vec{p} \equiv p_1, \dots, p_n$ of distinct propositional variables which includes all the variables that occur in ϕ, we associate the n-ary bit function

$$F^{\phi}_{p_1,\dots,p_n} : \mathbb{B}^n \to \mathbb{B}$$

by structural recursion, as follows.

(1) Prime formulas (propositional variables):

$$F^{p_i}_{p_1,\dots,p_n}(x_1, \dots, x_n) = x_i.$$

For example,

$$F^{\mathsf{A}_4}_{\mathsf{A}_4,\mathsf{A}_{15}}(x_1, x_2) = x_1, \quad F^{\mathsf{A}_4}_{\mathsf{A}_{12},\mathsf{A}_4,\mathsf{A}_7}(x_1, x_2, x_3) = x_2$$

and $F^{\mathsf{A}_4}_{\mathsf{A}_1,\mathsf{A}_{15}}$ is not defined because A_4 does occur in the list $\mathsf{A}_1, \mathsf{A}_{15}$.

(2) Negation:

$$F^{(\neg\psi)}_{\vec{p}}(\vec{x}) = 1 - F^{\psi}_{\vec{p}}(\vec{x}), = \begin{cases} 1, & \text{if } F^{\psi}_{\vec{p}}(\vec{x}) = 0, \\ 0, & \text{otherwise, i.e., if } F^{\psi}_{\vec{p}}(\vec{x}) = 1. \end{cases}$$

(3) Conjunction:

$$F_{\vec{p}}^{(\psi\wedge\chi)}(\vec{x}) = \min(F_{\vec{p}}^{\psi}(\vec{x}), F_{\vec{p}}^{\chi}(\vec{x})) = \begin{cases} 1, & \text{if } F_{\vec{p}}^{\psi}(\vec{x}) = 1 \text{ and } F_{\vec{p}}^{\chi}(\vec{x}) = 1, \\ 0, & \text{otherwise.} \end{cases}$$

(4) Disjunction:

$$F_{\vec{p}}^{(\psi\vee\chi)}(\vec{x}) = \max(F_{\vec{p}}^{\psi}(\vec{x}), F_{\vec{p}}^{\chi}(\vec{x})) = \begin{cases} 1, & \text{if } F_{\vec{p}}^{\psi}(\vec{x}) = 1 \text{ or } F_{\vec{p}}^{\chi}(\vec{x}) = 1, \\ 0, & \text{otherwise.} \end{cases}$$

(5) Implication:

$$F_{\vec{p}}^{(\psi\to\chi)}(\vec{x}) = \max(1 - F_{\vec{p}}^{\psi}(\vec{x}), F_{\vec{p}}^{\chi}(\vec{x})) = \begin{cases} 1, & \text{if } F_{\vec{p}}^{\psi}(\vec{x}) = 0 \text{ or } F_{\vec{p}}^{\chi}(\vec{x}) = 1, \\ 0, & \text{otherwise.} \end{cases}$$

The bit function $F_{\vec{p}}^{\phi}$ depends only on the propositional variables in the list $\vec{p}$ which actually occur in ϕ in the following sense:

2A.1. **Lemma.** *If $\vec{p}, q, \vec{r}$ is a sequence of distinct variables and q does not occur in a formula ϕ, then for all $\vec{x}, y, \vec{z} \in \mathbb{B}$,*

$$F_{\vec{p},q,\vec{r}}^{\phi}(\vec{x}, y, \vec{z}) = F_{\vec{p},\vec{r}}^{\phi}(\vec{x}, \vec{z})$$

We leave the proof for Problem x1.8.

2A.2. **Theorem** (Functional completeness of PL). *For every n-ary bit function $f : \mathbb{B}^n \to \mathbb{B}$, there exists a PL-formula ϕ and a sequence $\vec{p} \equiv p_1, \dots, p_n$ of distinct propositional variables which includes all the variables of ϕ such that*

$$f(x_1, \dots, x_n) = F_{\vec{p}}^{\phi}(x_1, \dots, x_n) \qquad (x_1, \dots, x_n \in \mathbb{B}).$$

Proof is by induction on n.

Basis, $n = 1$. There are only four unary bit functions, and each of them is defined by the formula in the table, relative to the variable p:

$f_1(x) = 1$	$p \vee \neg p$
$f_2(x) = 0$	$p \wedge \neg p$
$f_3(x) = x$	p
$f_4(x) = 1 - x$	$\neg p$

Induction step. Assume the result for n and suppose that f is $(n+1)$-ary. Consider the two functions obtained by fixing the last variable of f to be 1 or 0 and choose by the induction hypothesis formulas which define them relative to the variables $p_1, \dots, p_n$:

$f_1(x_1, \dots, x_n) = f(x_1, \dots, x_n, 1)$	defined by ϕ_1
$f_0(x_1, \dots, x_n) = f(x_1, \dots, x_n, 0)$	defined by ϕ_0

Now use Lemma 2A.1 to check that if p_{n+1} is a new propositional variable, then the formula

$$(p_{n+1} \wedge \phi_1\Big) \vee (\neg p_{n+1} \wedge \phi_0)$$

defines f relative to the list $p_1, \dots, p_n, p_{n+1}$. ⊣

Simple as it is, the Functional Completeness is the basis of many useful applications of Logic to Computer Science especially in the theory of circuits.

2B. Truth tables. There are 2^n n-tuples of 0's and 1's, and so the n-ary bit function defined by a formula relative to the variables $p_1, \dots, p_n$ can be pictured in a table with 2^n lines. For example, in the case of a formula with two variables (and including a column for the subformula $\neg p$ which is used in the computation):

p	q	$\neg p$	$\neg p \wedge q$
0	0	1	0
0	1	1	1
1	0	0	0
1	1	0	0

It is also useful to put down the following truth table which specifies succinctly the bit functions associated with all the connectives:

p	q	$\neg p$	$p \wedge q$	$p \vee q$	$p \rightarrow q$
0	0	1	0	0	1
0	1	1	0	1	1
1	0	0	0	1	0
1	1	0	1	1	1

2C. Satisfaction and the Tarski conditions. If we understand the propositional variables as standing for sentences with given *truth values* 1 or 0, then, directly from the definitions

$$F^{\phi}_{\vec{p}}(\vec{x}) = 1 \iff \phi \text{ is true when each } p_i \text{ has the truth value } x_i,$$

$$F^{\phi}_{\vec{p}}(\vec{x}) = 0 \iff \phi \text{ is false when each } p_i \text{ has the truth value } x_i.$$

To put this idea another way, consider **assignments** (to the variables), i.e., arbitrary functions

$$v : \{\mathbf{A}_0, \mathbf{A}_1, \dots, \} \rightarrow \mathbb{B}$$

which assign truth values to the propositional variables. For each formula ϕ, we set

$$\overline{v}(\phi) = F^{\phi}_{\vec{p}}(v(p_1), \dots, v(p_n))$$

where $\vec{p}$ is any list of distinct variables which includes all the variables that occur in ϕ. By Lemma 2A.1, the specific choice of the sequence of variables $p_1, \dots, p_n$ is immaterial, as long at it includes all the variables which occur in ϕ. We use the following terminology and notation for this important notion:

$$\begin{aligned} v \models \phi &\iff \overline{v}(\phi) = 1 \\ &\iff v \textbf{ satisfies } \phi \text{ or } \phi \textbf{ is true for the assignment } v. \end{aligned}$$

An assignment satisfies a (possibly infinite) set of formulas T if it satisfies every formula in T,

(2-1) $$v \models T \iff v \models \chi \text{ for every } \chi \in T;$$

and T is **satisfiable** if it is satisfied by some assignment v.

The only somewhat peculiar feature of this interpretation is for the conditional, for which it gives (for each fixed assignment)

$$(\phi \to \psi) \text{ is true } \iff \phi \text{ is false or } \psi \text{ is true,}$$

so that, for example, the sentence

if the moon is made of cheese, then I am 20 feet tall

comes out true. This is the *material implication* interpretation of conditionals and it is the most useful one for mathematics.

The satisfaction relation between assignments and formulas obeys the following classical rules which, in fact, determine it:

2C.1. **Theorem** (The Tarski conditions)**.** *For all variables p, formulas ϕ, ψ and assignments v:*

$$\begin{aligned} v \models p &\iff v(p) = 1, \\ v \models \neg\phi &\iff v \not\models \phi, \\ v \models \phi \wedge \psi &\iff v \models \phi \text{ and } v \models \psi, \\ v \models \phi \vee \psi &\iff v \models \phi \text{ or } v \models \psi, \\ v \models \phi \to \psi &\iff \text{either } v \not\models \phi \text{ or } v \models \psi. \end{aligned}$$

We leave the easy proof for Problem x1.9.

2D. Tautologies and logical consequence. A tautology is a formula whose associated bit function is the constant 1, i.e., every assignment satisfies it. We write

(2-2) $$\models \phi \iff \phi \text{ is a tautology } \iff \text{ for every assignment } v, v \models \phi.$$

More generally, for any set T of formulas and any formula ϕ, we set

(2-3) $$\begin{aligned} T \models \phi \iff &\text{ for every assignment } v, \\ &\text{if } v \text{ satisfies all the formulas in } T, \text{ then } v \text{ also satisfies } \phi. \end{aligned}$$

If $T \models \phi$, we say that ϕ is a **logical consequence** of T.

Notational conventions: We write

$$\phi_1, \dots, \phi_n \models \phi \iff \{\phi_1, \dots, \phi_n\} \models \phi,$$
$$T, \phi_1, \dots, \phi_n \models \phi \iff T \cup \{\phi_1, \dots, \phi_n\} \models \phi.$$

In particular,

$$\models \phi \iff \emptyset \models \phi,$$

and this agrees with our notation above and exhibits the tautologies as the logical consequences of the empty set of assumptions.

Notice that this "sequential notation" for sets allows repetitions and reordering, e.g.,

$$\phi, \phi, \psi \models \chi \iff \{\phi, \phi, \psi\} \models \chi \iff \{\phi, \psi\} \models \chi \iff \psi, \phi \models \chi.$$

This is convenient when we have a list of formulas given in no particular order and we do not know which of them may be equal to some others.

Replacement. Suppose ϕ is a formula, $p_1, \dots, p_k$ are distinct variables which may or may not occur in ϕ, and $\psi_1, \dots, \psi_k$ is a sequence of formulas. We set

$$\begin{aligned}&\phi\{p_1 :\equiv \psi_1, \dots, p_k :\equiv \psi_k\}\\ &\qquad = \text{the result of replacing each occurrence of each } p_i \text{ in } \phi \text{ by } \psi_i.\end{aligned}$$

For example:

$$p\{p :\equiv \psi, q :\equiv \chi\} \equiv \psi$$
$$p \to (q \to p)\{p :\equiv \psi, q :\equiv \chi\} \equiv \psi \to (\chi \to \psi)$$
$$p \wedge (p \to q)\{p :\equiv \psi, q :\equiv \chi\} \equiv \psi \wedge (\psi \to \chi)$$

2D.1. **Theorem** (The Replacement Theorem). *If ϕ is a tautology, then the result $\phi\{p_1 :\equiv \psi_1, \dots, p_k :\equiv \psi_k\}$ of any simultaneous replacement on ϕ is also a tautology.*

Proof. We may assume that the sequence $\vec{p} \equiv p_1, \dots, p_k$ includes all the variables which occur in ϕ, by adding trivial replacements of the form $p_i :\equiv p_i$ if necessary. Let $\vec{q} \equiv q_1, \dots, q_l$ be a sequence of distinct variables which include all the variables in $\psi_1, \dots, \psi_k$ and let

$$\chi \equiv \phi\{p_1 :\equiv \psi_1, \dots, p_k :\equiv \psi_k\}$$

be the result of the replacement, so that the list $\vec{q}$ includes all the variables which occur in χ. With these notation conventions and $\vec{y} = (y_1, \dots, y_l)$, we can check that

$$F^{\chi}_{\vec{q}}(\vec{y}) = F^{\phi}_{\vec{p}}(F^{\psi_1}_{\vec{q}}(\vec{y}), \dots, F^{\psi_k}_{\vec{q}}(\vec{y})), \tag{2-4}$$

(cf. Problem x1.10). This basic equation implies the theorem immediately, since when ϕ is a tautology, then $F^{\phi}_{\vec{p}}$ is the constant function with value 1—and hence so is $F^{\chi}_{\vec{q}}$. ⊣

Tautologies are easily recognized by inspection of their truth tables, which must have only 1's in the column under the formula—except that it is very tedious to construct truth tables. An easier method is to use the definition (2-2) and the Tarski conditions in Theorem 2C.1, as in this

2D.2. **Lemma.** *For all formulas ϕ and ψ, $\models \phi \to (\psi \to \phi)$.*

Proof. Following the hint above, we compute:

$$\begin{aligned}\models \phi \to (\psi \to \phi) &\iff \text{for all } v, v \models \phi \to (\psi \to \phi)\\ &\iff \text{for all } v, \text{ either } v \not\models \phi \text{ or } v \models (\psi \to \phi)\\ &\iff \text{for all } v, \text{ either } v \not\models \phi \text{ or (either } v \not\models \psi \text{ or } v \models \phi)\\ &\iff \text{for all } v, \text{ either } v \not\models \phi \text{ or } v \not\models \psi \text{ or } v \models \phi\end{aligned}$$

and the condition on the last line is obviously true for every v. ⊣

The formula $\phi \to (\psi \to \phi)$ is the first in the following list of tautologies that we will find useful in the next section:

2D.3. **Theorem.** *For all formulas ϕ, ψ, χ,*

(1) $\models \phi \to (\psi \to \phi)$
(2) $\models (\phi \to \psi) \to ((\phi \to (\psi \to \chi)) \to (\phi \to \chi))$
(3) $\models (\phi \to \psi) \to ((\phi \to \neg\psi) \to \neg\phi)$
(4) $\models \neg\neg\phi \to \phi$
(5) $\models \phi \to (\psi \to (\phi \wedge \psi))$
(6a) $\models (\phi \wedge \psi) \to \phi$
(6b) $\models (\phi \wedge \psi) \to \psi$
(7a) $\models \phi \to (\phi \vee \psi)$
(7b) $\models \psi \to (\phi \vee \psi)$
(8) $\models (\phi \to \chi) \to ((\psi \to \chi) \to ((\phi \vee \psi) \to \chi))$

These are all quite easy to check as we proved Lemma 2D.2, cf. Problems x1.11, x1.12.

2D.4. **Lemma** (Modus Ponens). *For any two formulas ϕ, ψ:*

$$\phi, \phi \to \psi \models \psi.$$

It follows that for any set T of assumptions and any ϕ, ψ,

$$\text{if } T \models \phi \text{ and } T \models \phi \to \psi, \text{ then } T \models \psi.$$

Proof is left for Problem x1.13. ⊣

§3. Formal deduction. Intuitively, a **proof** of a claim C is a justification of the truth of C on the basis of some *logical axioms* (which we take to be self-evident) and some *rules of inference* for which it is evident that they preserve (respect) truth. Similarly with a **deduction** or **proof** of C **from a set** T **of assumptions**: it should "certify" that every interpretation which makes all the assumptions in T true also makes C true. Here we make these notions precise for the propositional calculus, for which the "claims" and the "assumptions" are propositional formulas.

3A. Axioms and proofs for PL**.** The **Hilbert axioms** for PL are the formulas (1) – (8) in Theorem 2D.3; there are infinitely many of them since ϕ, ψ, χ stand for arbitrary formulas, and so, more properly we should refer to (1) – (8) as **axiom schemes**.

There is only one **rule of inference** in PL,

Modus Ponens: from ϕ and $\phi \to \psi$, infer ψ.

A (propositional) **deduction** or **proof from** a set of formulas T is any sequence of formulas

$$\chi_0, \chi_1, \dots, \chi_k$$

such for each $n \leq k$ one of the following holds:

(D1) $\chi_n \in T$ (assumption).
(D2) $\chi_n \equiv \chi_i$ for some $i < n$ (repetition).
(D3) χ_n is an axiom.
(D4) χ_n can be inferred with Modus Ponens from some χ_i, χ_j with $i, j < n$.

We set

$$\boxed{T \vdash \chi \iff \text{there exists a proof } \chi_0, \chi_1, \dots, \chi_k \text{ from } T \text{ such that } \chi \equiv \chi_k}$$

and (as with $\models$), we just list T if it is finite and we skip it entirely when it is empty,

$$T, \phi_1, \dots, \phi_m \vdash \chi \iff T \cup \{\phi_1, \dots, \phi_m\} \vdash \chi, \quad \vdash \chi \iff \emptyset \vdash \chi.$$

If $T \vdash \chi$, we say that T **proves** χ or χ is **deducible** from T and we call χ a **theorem of** T.

Combining deductions. If $\chi_1, \dots, \chi_k$ and $\psi_1, \dots, \psi_m$ are deductions from T, then their concatenation

$$\chi_1, \dots, \chi_k, \psi_1, \dots, \psi_m$$

is also a deduction from T, and a deduction from $T \cup S$ for any S. These are two of several trivial properties of deductions which we will use, often without explicit mention.

A set T of formulas is **deductively closed** if it contains all the axioms and is closed under Modus Ponens, i.e.,

$$\phi, \phi \to \psi \in T \Longrightarrow \psi \in T.$$

3A.1. **Lemma.** *For every T and every ϕ,*

$$T \vdash \phi \iff \phi \in S \text{ for every deductively closed } S \supseteq T$$

This is proved exactly like Problem x1.2 and we leave it for Problem x1.16.

3A.2. **Lemma.** *For every formula ϕ, $\vdash \phi \to \phi$.*

Proof. Here is a fully annotated proof of this obvious tautology:

1. $(\phi \to (\phi \to \phi)) \to ((\phi \to ((\phi \to \phi) \to \phi)) \to (\phi \to \phi))$
   Taking $\psi \equiv (\phi \to \phi)$ and $\chi \equiv \phi$ in Axiom Scheme (2).
2. $\phi \to (\phi \to \phi)$
   Taking $\psi \equiv \phi$ in Axiom Scheme (1).
3. $(\phi \to ((\phi \to \phi) \to \phi)) \to (\phi \to \phi)$
   By Modus Ponens on 1 and 2.
4. $\phi \to ((\phi \to \phi) \to \phi)$
   Taking $\psi \equiv \phi \to \phi$ in Axiom Scheme (1).
5. $\phi \to \phi$
   By Modus Ponens on 4 and 3. ⊣

3A.3. **The Deduction Theorem for PL.** *For any set of formulas T and all ϕ, ψ, if $T, \phi \vdash \psi$, then $T \vdash (\phi \to \psi)$.*

Proof. Let $\chi_0, \dots, \chi_k$ be the assumed deduction from $T \cup \{\phi\}$ with $\psi \equiv \chi_k$. It is enough to show that $T \vdash (\phi \to \chi_n)$ for every $n \leq k$, and we do this by (complete) induction on $n \leq k$.

If $\chi_n \equiv \phi$, then $T \vdash \phi \to \phi$ by Lemma 3A.2, and if $\chi_n \equiv \chi_i$ for some $i < n$ then the induction hypothesis gives $T \vdash \phi \to \chi_n$.

If χ_n is an axiom or in T, then the following is a deduction of $\phi \to \chi_n$ from T using Axiom Scheme (1) and Modus Ponens:

$$\chi_n,\ \chi_n \to (\phi \to \chi_n),\ \phi \to \chi_n$$

Finally, if χ_n is inferred by Modus Ponens from previously listed formulas χ_i, χ_j, then $\chi_j \equiv \chi_i \to \chi_n$ and the induction hypothesis gives us deductions from T of $\phi \to \chi_i$ and $\phi \to (\chi_i \to \chi_n)$; we construct a deduction of $\phi \to \chi_n$ from T starting with these and continuing using Axiom Scheme (2) and two more applications of Modus Ponens as follows:

$$\begin{aligned}
\text{from } T : \dots, \phi \to \chi_i, \dots, &\phi \to (\chi_i \to \chi_n),\\
&(\phi \to \chi_i) \to ((\phi \to (\chi_i \to \chi_n)) \to (\phi \to \chi_n)),\\
&(\phi \to (\chi_i \to \chi_n)) \to (\phi \to \chi_n),\\
&\phi \to \chi_n
\end{aligned}$$

⊣

The Deduction Theorem is an example of a **metatheorem** for PL, a theorem about formal proofs which can be used to show that formal proofs exist without actually constructing them. We list in the next two lemmas some of the most useful metatheorems for PL, leaving the (quite easy) proofs for problems.

3A.4. **Lemma** (The natural introduction rules for PL). *For every set of formulas T:*

($\to$) *If $T, \chi \vdash \phi$, then $T \vdash \chi \to \phi$.*
($\wedge$) *If $T \vdash \phi$ and $T \vdash \psi$, then $T \vdash \phi \wedge \psi$.*
($\vee$) *If $T \vdash \phi$ or $T \vdash \psi$, then $T \vdash \phi \vee \psi$.*
($\neg$) *If $T, \chi \vdash \psi$ and $T, \chi \vdash \neg\psi$, then $T \vdash \neg\chi$.*

3A.5. **Lemma** (The natural elimination rules for PL). *For every set of formulas T:*

($\to$) *If $T \vdash \phi$ and $T \vdash \phi \to \psi$, then $T \vdash \psi$.*
($\wedge$) *If $T \vdash \phi \wedge \psi$, then $T \vdash \phi$ and $T \vdash \psi$.*
($\vee$) *If $T, \phi \vdash \chi$ and $T, \psi \vdash \chi$, then $T, \phi \vee \psi \vdash \chi$*
($\neg$) *If $T \vdash \neg\neg\phi$, then $T \vdash \phi$.*

3B. Soundness and Completeness of PL. We now turn to the two basic results which relate the syntax and the semantics of PL:

3B.1. **Theorem** (Soundness). *For any set of formulas T and every formula ϕ,*

$$\textit{if } T \vdash \phi\ ,\ \textit{then } T \models \phi.$$

Proof. The set $L(T)$ of logical consequences of T includes T and is deductively closed, because every axiom is a tautology by Theorem 2D.3 and the rule of Modus Ponens preserves logical consequence, by Lemma 2D.4. By Lemma 3A.1 then, $L(T)$ contains all the theorems of T. ⊣

For the converse result we need two basic notions:

3B.2. **Definition** (Consistency and strong completeness). A set of formulas T is **consistent** if there is no formula ϕ such that

$$T \vdash \phi \text{ and } T \vdash \neg\phi;$$

and T is **strongly complete** if for every formula ϕ,

$$\text{either } \phi \in T \text{ or } \neg\phi \in T.$$

3B.3. **Lemma.** *A set of formulas S is consistent and strongly complete if and only if there is an assignment v to the propositional variables such that for every formula ϕ,*

$$v \models \phi \iff \phi \in S. \tag{3-1}$$

Proof. If (3-1) holds for S and some v, then S is obviously strongly complete since for every ϕ, either $v \models \phi$ or $v \models \neg\phi$. To see that it is also consistent, notice that by the Soundness Theorem 3B.1,

$$S \vdash \phi \Longrightarrow S \models \phi \Longrightarrow v \models \phi,$$

which with (3-1) eliminates the possibility that for some ϕ, $S \vdash \phi$ and also $S \vdash \neg\phi$.

For the converse, we assume that S is consistent and strongly complete. Notice first that that S is *closed under deduction*, i.e.,

$$S \vdash \phi \Longrightarrow \phi \in S; \tag{3-2}$$

because if $S \vdash \phi$ but $\phi \notin S$, then $\neg\phi \in S$ by the strong completeness, and so $S \vdash \neg\phi$ which makes it inconsistent. With this in mind, set

$$v(p) = 1 \iff p \in S$$

and prove (3-1) by induction on ϕ. It is true at the basis (when ϕ is a propositional variable) by the definition of v, and to consider just two of the cases at the induction step:

$$\begin{aligned} v \models \neg\phi &\iff v \not\models \phi \iff \phi \notin S \text{ (ind. hyp)} \\ &\qquad\qquad \iff \neg\phi \in S \text{ (strong completeness)}, \\ v \models \phi \wedge \psi &\iff v \models \phi \text{ and } v \models \psi \iff \phi \in S \text{ and } \psi \in S \\ &\qquad\qquad \iff \phi \wedge \psi \in S \end{aligned}$$

the last step by (3-2), since

$$\phi, \psi \vdash \phi \wedge \psi \text{ and } \phi \wedge \psi \vdash \phi, \phi \wedge \psi \vdash \psi. \qquad \dashv$$

3B.4. **Lemma.** *If T is a consistent set of formulas and χ is any formula, then*

$$\textit{either } T \cup \{\chi\} \textit{ is consistent or } T \cup \{\neg\chi\} \textit{ is consistent.}$$

Proof. If $T \cup \{\chi\}$ is inconsistent, then there is a formula ϕ such that

$$T, \chi \vdash \phi \text{ and } T, \chi \vdash \neg\phi,$$

so that by the ($\neg$)-introduction rule in Lemma 3A.4,

$$T \vdash \neg\chi;$$

and if $T \cup \{\neg\chi\}$ is also inconsistent, by the same argument, $T \vdash \neg\neg\chi$, which by the ($\neg$)-elimination rule in Lemma 3A.5 gives

$$T \vdash \chi;$$

but now the last two displays imply that T is inconsistent, contrary to the hypothesis. $\dashv$

3B.5. **Theorem** (The Completeness Theorem for PL)**.** (1) *Every consistent set of formulas is satisfiable.*

(2) *For every set of formulas T and every χ,*

$$\text{if } T \models \chi, \text{ then } T \vdash \chi.$$

PROOF. (1) The idea is to extend T by adding to it successively each formula or its negation, using the preceding Lemma 3B.4, until we get in the end a consistent, strongly complete set of formulas $T^* \supseteq T$ which has a satisfying assignment by Lemma 3B.3.

In detail, let F_n be the set of formulas of length $\leq n+1$ in which only the variables $\mathbf{A}_0, \ldots, \mathbf{A}_n$ may occur, e.g., $F_0 = \{\mathbf{A}_0\}$. This set is finite, for every n, so we can fix an enumeration

$$F_n = \{\chi_0^n, \ldots, \chi_{k_n}^n\}$$

of it. Lining up these enumerations in sequence, one after another, we obtain an enumeration (with many irrelevant repetitions)

$$\chi_0, \chi_1, \ldots$$

of all PL-formulas, which we fix. We now define a sequence of sets

$$T_0 \subseteq T_1 \subseteq T_2 \cdots$$

by setting recursively

$$T_0 = T, \quad T_{n+1} = \begin{cases} T_n \cup \{\chi_n\} & \text{if } T_n \cup \{\chi_n\} \text{ is consistent,} \\ T_n \cup \{\neg\chi_n\} & \text{otherwise,} \end{cases}$$

so that (inductively) each T_n is consistent, so $T^* = \bigcup_n T_n \supseteq T$ is also consistent by Problem x1.24, and it is obviously strongly complete.

(2) Assume, towards a contradiction that

$$T \models \chi \text{ but } T \nvdash \chi.$$

It follows that the set $T \cup \{\neg\chi\}$ is consistent, since if

$$T, \neg\chi \vdash \phi \text{ and } T, \neg\chi \vdash \neg\phi,$$

then $T \vdash \neg\neg\chi$ and hence $T \vdash \chi$, as above, which contradicts the hypothesis. So there is an assignment v which satisfies every formula in $T \cup \{\neg\chi\}$ by (1), contradicting the assumption $T \models \chi$. $\dashv$

It is Part (2) which is properly called the Completeness Theorem, but Part (1) is the version of it which is more useful for many applications.

§4. Problems.

x1.1. Prove that the set of propositional formulas defined in 1A.1 is the smallest set of strings which is propositionally closed; i.e., it is propositionally closed and it is a subset of every propositionally closed set of strings.

x1.2. A sequence $\alpha_0, \ldots, \alpha_k$ of strings is a *propositional formula construction* if for each $n \le k$, either α_n is a propositional variable, or $\alpha_n \equiv (\neg \alpha_i)$ for some $i < n$, or $\alpha_n \equiv (\alpha_i \bullet \alpha_j)$ for some binary connective $\bullet$ and some $i, j < n$.

Prove that a string of symbols ϕ is a propositional formula if and only if it occurs in some propositional formula construction.

x1.3. Prove (2) of Lemma 1A.2, that for every non-empty, proper, initial part α of a formula ϕ, the number of left parentheses "(" in α is greater than the number of right parentheses ")" in α.

x1.4* (Parsing Lemma for PL). Prove Theorem 1A.3. Hint: For the most interesting part (3) of the uniqueness claim, suppose

$$(\psi_1 \bullet_1 \chi_1) \equiv (\psi_2 \bullet_2 \chi_2)$$

with ψ_1 shorter than ψ_2 and derive a contradiction using Lemma 1A.2.

x1.5. For each of the following equations between strings, determine whether there are formulas $\phi, \psi, \chi, \phi', \psi', \chi'$ which make them true:

(a) $((\phi \wedge \psi) \vee \chi) \equiv (\phi' \vee (\psi' \wedge \chi'))$.

(b) $((\phi \wedge \psi) \vee \chi) \equiv (\phi' \wedge (\psi' \vee \chi'))$.

You must prove your answers.

x1.6. Construct the truth table for the formula $(p \to q) \wedge \neg(q \to p)$.

x1.7. Let $\downarrow$ be the *Sheffer stroke*, the binary connective defined by the truth table

p	q	$p \downarrow q$
0	0	1
0	1	0
1	0	0
1	1	0

We read $(\phi \downarrow \psi)$ as "*neither* ϕ *nor* ψ". Define the $\downarrow$-formulas using only this connective (rather than $\neg, \wedge, \vee, \to, \leftrightarrow$), and prove that every n-ary bit function can be defined by a $\downarrow$-formula with n propositional variables.

x1.8. Prove Lemma 2A.1.

x1.9. Prove the Tarski conditions, Theorem 2C.1.

x1.10. Prove equation (2-4) in the proof of the Replacement Theorem 2D.1, i.e., the following: for every formula ϕ whose variables are in the list $p_1, \dots, p_k$, if $\vec{q} \equiv q_1, \dots, q_l$ is a sequence of distinct variables which include all the variables in $\psi_1, \dots, \psi_k$, $\vec{y} \equiv y_1, \dots, y_l$ and

$$\chi \equiv \phi\{p_1 :\equiv \psi_1, \dots, p_k :\equiv \psi_k\},$$

then

$$(*) \qquad F^{\chi}_{\vec{q}}(\vec{y}) = F^{\phi}_{\vec{p}}(F^{\psi_1}_{\vec{q}}(\vec{y}), \dots, F^{\psi_k}_{\vec{q}}(\vec{y})).$$

Hint: Use structural induction on ϕ.

x1.11. Prove that the formula (2) in Theorem 2D.3 is a tautology.

x1.12. Prove that the formula (8) in Theorem 2D.3 is a tautology.

x1.13. Prove Lemma 2D.4, the Modus Ponens Rule.

x1.14. Prove that

$$\text{if } T, \phi \models \psi, \text{ then } T \models \phi \to \psi.$$

4.1. **Definition.** For any finite sequence of formulas $\chi_0, \chi_1, \dots, \chi_n$, set

$$\textstyle\bigvee\!\!\!\bigvee_{i \le n} \chi_i :\equiv \chi_0 \vee \chi_1 \vee \cdots \vee \chi_n$$

so that, for example

$$\textstyle\bigvee\!\!\!\bigvee_{i \le 0} \chi_i \equiv \chi_0, \ \bigvee\!\!\!\bigvee_{i \le 1} \chi_i \equiv \chi_0 \vee \chi_1, \ \bigvee\!\!\!\bigvee_{i \le 2} \chi_i \equiv \chi_0 \vee \chi_1 \vee \chi_2.$$

Similarly for abbreviation of finite conjunctions:

$$\textstyle\bigwedge\!\!\!\bigwedge_{i \le n} \chi_i :\equiv \chi_0 \wedge \chi_1 \wedge \cdots \wedge \chi_n$$

x1.15*. Suppose $R(i, j)$ is a relation defined for $i, j \le n$, choose a doubly indexed sequence of distinct propositional variables $\{p_{ij}\}_{i,j \le n}$, and consider the assignment

$$v(p_{ij}) = \begin{cases} 1, & \text{if } R(i, j), \\ 0, & \text{otherwise.} \end{cases}$$

The variables $\{p_{ij}\}$ can be used to express various properties about the relation R. Recall for example that

$$R \text{ is symmetric} \iff (\text{for all } i, j \le n)[R(i, j) \iff R(j, i)];$$

now easily,

$$R \text{ is symmetric} \iff v \models \textstyle\bigwedge\!\!\!\bigwedge_{i \le n} \bigwedge\!\!\!\bigwedge_{j \le n} [p_{ij} \leftrightarrow p_{ji}].$$

Find propositional formulas which express the following properties of R:

(a) R is the graph of a function, i.e., $(R(i, j) \ \& \ R(i, k)) \Longrightarrow j = k$.
(b) R is the graph of a one-to-one function.
(c) R is the graph of a surjection—a function from $\{0, \dots, n\}$ onto $\{0, \dots, n\}$.

x1.16. Prove Lemma 3A.1, that for every T and every ϕ,

$$T \vdash \phi \iff \phi \in S \text{ for every deductively closed } S \supseteq T$$

x1.17. Prove that the "repetition" clause (D2) in the definition of deduction is not needed: i.e., *if* $T \vdash \chi$, *then there is a deduction of* χ *from* T *without repetitions*. (The clause was included in the definition so that we can more easily combine deductions without restriction.)

x1.18. Prove the ($\wedge$)-introduction rule in Lemma 3A.4: that

$$\text{if } T \vdash \phi \text{ and } T \vdash \psi, \text{ then } T \vdash (\phi \wedge \psi).$$

x1.19. Prove the ($\neg$)-introduction rule in Lemma 3A.4: that

$$\text{if } T, \chi \vdash \phi \text{ and } T, \chi \vdash \neg\phi, \text{ then } T \vdash \neg\chi.$$

x1.20. Prove the ($\vee$)-elimination rule in Lemma 3A.5, that

$$\text{if } T, \phi \vdash \chi \text{ and } T, \psi \vdash \chi, \text{ then } T, \phi \vee \psi \vdash \chi.$$

x1.21. Prove the ($\neg$)-elimination rule in Lemma 3A.5, that

$$\text{if } T \vdash \neg\neg\phi, \text{ then } T \vdash \phi.$$

x1.22. Prove that if T is not consistent, then $T \vdash \chi$ for every χ.

x1.23* (The law of excluded middle). Prove that for every formula χ,

$$\vdash \chi \vee \neg\chi.$$

HINT: Prove $\neg\neg(\chi \vee \neg\chi)$ and then use Axiom (4), or use the natural introduction and elimination rules in Lemmas 3A.4, 3A.5.

x1.24. Prove that a set of formulas T is consistent if and only if every finite subset of T is consistent.

x1.25*. Prove *Peirce's Law*, the formula

$$(((p \to q) \to p) \to p)$$

Note: It is trivial to check that Peirce's Law is valid, i.e., every assignment v satisfies it: just take cases on whether $v \models p$ or $v \models \neg p$. The challenge is to give a proof and check which axioms or which natural introduction and elimination rules are needed.

x1.26. Two formulas ϕ and ψ are (logically) *equivalent* if $\phi \models \psi$ and $\psi \models \phi$. Prove that for each sequence $p_1, \ldots, p_n$ of distinct propositional variables, there is a sequence

$$\chi_1, \ldots, \chi_N \text{ with } N = 2^{2^n}$$

of inequivalent formulas in the variables $p_1, \ldots, p_n$, such that every formula in which only these variables occur is equivalent to some χ_i. (For example, when $n = 1$, the required sequence is $p, \neg p, p \wedge p, p \vee \neg p$.) HINT: Count truth tables.

x1.27 (The Compactness Theorem for PL). Suppose T is an infinite set of formulas. Prove that if every finite subset $T_0 \subset T$ of T is satisfiable, then T is satisfiable.