Introduction to Logic for Computer Science

S. Arun-Kumar

September 26, 2002

Contents

Chapter 1

Introduction and Mathematical Preliminaries

> **al-go-rism** *n. [ME* algorsme*<OFr.< Med.Lat.* algorismus*, after Muhammad ibn-Musa Al-Kharzimi (780-850?).] The Arabic system of numeration:* DECIMAL SYSTEM.
>
> **al-go-rithm** *n [Var. of* ALGORISM.*] Math. A mathematical rule or procedure for solving a problem.*
> △*word history: Algorithm originated as a varaiant spelling of* algorism*. The spelling was probably influenced by the word* aruthmetic *or its Greek source* arithm*, "number". With the development of sophisticated mechanical computing devices in the 20th century, however,* algorithm *was adopted as a convenient word for a recursive mathematical procedure, the computer's stock in trade.* Algorithm *has ceased to be used as a variant form of the older word.*
>
> Webster's II New Riverside University Dictionary 1984.

1.1 Motivation for the Study of Logic

In the early years of this century symbolic or formal logic became quite popular with philosophers and mathematicicans because they were interested in the concept of what constitutes a correct proof in mathematics. Over the centuries mathematicians had pronounced various mathematical proofs as correct which were later disproved by other mathematicians. The whole concept of logic then hinged upon what is a correct argument as opposed to a wrong (or faulty) one. This has been amply illustrated by the number of so-called proofs that have come up for Euclid's parallel postulate and for Fermat's last theorem. There have invariably been "bugs" (a term popularised by computer scientists for the faults in a program) which were often very hard to detect and it was necessary therefore to find infallible methods of proof. For centuries (dating back at least to Plato and Aristotle) no rigorous formulation was attempted to capture the notion of a correct argument which would guide the development of all mathematics.

The early logicians of the nineteenth and twentieth centuries hoped to establish formal logic as a foundation for mathematics, though that never really happened. But mathematics does rest on one firm foundation, namely set theory. But Set theory itself has been expressed in first order logic. What really needed to be answered were questions relating to the automation or mechanizability of proofs. These questions are very relevant and important for the development of present-day computer science and form the basis of many developments in automatic theorem proving. David Hilbert asked the important question, as to whether all mathematics, if reduced to statements of symbolic logic, can be derived by a machine. Can the act of constructing a proof be reduced to the manipulation of statements in symbolic logic? Logic enabled mathematicians to point out why an alleged proof is wrong, or where in the proof, the reasoning has been faulty. A large part of the credit for this achievement must go to the fact that by symbolising arguments rather than writing them out in some natural language (which is fraught with ambiguity), checking the correctness of a proof becomes a much more viable task. Of course, trying to symbolise the whole of mathematics could be disastrous as then it would become quite impossible to even read and understand mathematics, since what is presented usually as a one page proof could run into several pages. But at least in principle it can be done.

Since the latter half of the twentieth century logic has been used in computer science for various purposes ranging from program specification and verification to theorem-proving. Initially its use was restricted to merely specifying programs and reasoning about their implementations. This is exemplified in the some fairly elegant research on the development of correct programs using first-order logic in such calculi such as the weakest-precondition calculus of Dijkstra. A method called Hoare Logic which combines first-order logic sentences and program phrases into a specification and reasoning mechanism is also quite useful in the development of small programs. Logic in this form has also been used to specify the meanings of some programming languages, notably Pascal.

The close link between logic as a formal system and computer-based theorem proving is proving to be very useful especially where there are a large number of cases (following certain patterns) to be analysed and where quite often there are routine proof techniques available which are more easily and accurately performed by therorem-provers than by humans. The case of the four-colour theorem which until fairly recently remained a unproved conjecture is an instance of how human ingenuity and creativity may be used to divide up proof into a few thousand cases and where machines may be used to perform routine checks on the individual cases. Another use of computers in theorem-proving or model-checking is the verification of the design of large circuits before a chip is fabricated. Analysing circuits with a billion transistors in them is at best error-prone and at worst a drudgery that few humans would like to do. Such analysis and results are best performed by machines using theorem proving techniques or model-checking techniques.

A powerful programming paradigm called declarative programming has evolved since the late seventies and has found several applications in computer science and artificial intelligence. Most programmers using this logical paradigm use a language called Prolog which is an implemented form of logic[1]. More recently computer scientists are working on a form of logic called constraint logic programming.

In the rest of this chapter we will discuss sets, relations, functions. Though most of these topics are covered in the high school curriculum this section also establishes the notational conventions that will be used throughout. Even a confident reader may wish to browse this section to get familiar with the notation.

1.2 Sets

A *set* is a collection of *distinct* objects. The class of CS253 is a set. So is the group of all first year students at the IITD. We will use the notation $\{a, b, c\}$ to denote the collection of the objects a, b and c. The elements in a set are not ordered in any fashion. Thus the set $\{a, b, c\}$ is the same as the set $\{b, a, c\}$. Two sets are *equal* if they contain exactly the same elements.

We can describe a set either by enumerating all the elements of the set or by stating the properties that uniquely characterize the elements of the set. Thus, the set of all even positive integers not larger than 10 can be described either as $S = \{2, 4, 6, 8, 10\}$ or, equivalently, as $S = \{x \mid x \text{ is an even positive integer not larger than } 10\}$

A set can have another set as one of its elements. For example, the set $A = \{\{a, b, c\}, d\}$ contains two elements $\{a, b, c\}$ and d; and the first element is itself a set. We will use the notation $x \in S$ to denote that x is an *element of* (or *belongs to*) the set S.

A set A is a *subset* of another set B, denoted as $A \subseteq B$, if $x \in B$ whenever $x \in A$.

An *empty set* is one which contains no elements and we will denote it with the symbol $\emptyset$. For example, let S be the set of all students who fail this course. S might turn out to be empty (hopefully; if everybody studies hard). By definition, the empty set $\emptyset$ is a subset of all sets. We will also assume an *Universe of discourse* $\mathbb{U}$, and every set that we will consider is a subset of $\mathbb{U}$. Thus we have

1. $\emptyset \subseteq A$ for any set A
2. $A \subseteq \mathbb{U}$ for any set A

The *union* of two sets A and B, denoted $A \cup B$, is the set whose elements are exactly the elements of either A or B (or both). The *intersection* of two sets A and B, denoted $A \cap B$, is the set whose elements are exactly the elements that belong to *both* A and B. The *difference* of B from A, denoted $A - B$, is the set of all elements of A that do not belong to B. The *complement* of A, denoted $\sim A$ is the difference of A from the universe $\mathbb{U}$. Thus, we have

1. $A \cup B = \{x \mid (x \in A) \text{ or } (x \in B)\}$

[1] actually a subset of logic called Horn-clause logic

2. $A \cap B = \{x \mid (x \in A) \text{ and } (x \in B)\}$
3. $A - B = \{x \mid (x \in A) \text{ and } (x \notin B)\}$
4. $\sim A = \mathbb{U} - A$

We also have the following named identities that hold for all sets A, B and C.

Basic properties of set union.

1. $(A \cup B) \cup C = A \cup (B \cup C)$ *Associativity*
2. $A \cup \phi = A$ *Identity*
3. $A \cup \mathbb{U} = \mathbb{U}$ *Zero*
4. $A \cup B = B \cup A$ *Commutativity*
5. $A \cup A = A$ *Idempotence*

Basic properties of set intersection

1. $(A \cap B) \cap C = A \cap (B \cap C)$ *Associativity*
2. $A \cap \mathbb{U} = A$ *Identity*
3. $A \cap \phi = \phi$ *Zero*
4. $A \cap B = B \cap A$ *Commutativity*
5. $A \cap A = A$ *Idempotence*

Other properties

1. $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$ *Distributivity of* $\cap$ *over* $\cup$
2. $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ *Distributivity of* $\cup$ *over* $\cap$
3. $\sim (A \cup B) =\sim A \cap \sim B$ *De Morgan's law* $\sim \cup$
4. $\sim (A \cap B) =\sim A \cup \sim B$ *De Morgan's law* $\sim \cap$
5. $A \cap (\sim A \cup B) = A \cap B$ *Absorption* $\cup$
6. $A \cup (\sim A \cap B) = A \cup B$ *Absorption* $\cap$

The reader is encouraged to come up with properties of set difference and the complementation operations.

We will use the following notation to denote some standard sets:

The empty set: $\emptyset$

The Universe: $\mathbb{U}$

The Powerset of a set A: $\mathbf{2}^A$ is the set of all subsets of the set A.

The set of Natural Numbers: [2] $\mathbb{N} = \{0, 1, 2, \ldots\}$

The set of positive integers: $\mathbb{P} = \{1, 2, 3, \ldots\}$

The set of integers: $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$

The set of real numbers: $\mathbb{R}$

The Boolean set: $\mathbb{B} = \{false, true\}$

1.3 Relations and Functions

The *Cartesian product* of two sets A and B, denoted by $A \times B$, is the set of all ordered pairs (a, b) such that $a \in A$ and $b \in B$. Thus,

$$A \times B = \{(a, b) \mid a \in A \text{ and } b \in B\}$$

Given another set C we may form the following different kinds of cartesian products (which are not at all the same!).

$$(A \times B) \times C = \{((a, b), c) \mid a \in A, b \in B \text{ and } c \in C\}$$

$$A \times (B \times C) = \{(a, (b, c)) \mid a \in A, b \in B \text{ and } c \in C\}$$

$$A \times B \times C = \{(a, b, c) \mid a \in A, b \in B \text{ and } c \in C\}$$

The last cartesian product gives the construction of tuples. Elements of the set $A_1 \times A_2 \times \cdots \times A_n$ for given sets $A_1, A_2, \ldots, A_n$ are called *ordered n-tuples*.

[2] We will include 0 in the set of Natural numbers. After all, it is quite natural to score a 0 in an examination

A^n is the set of all ordered n-tuples $(a_1, a_2, \ldots, a_n)$ such that $a_i \in A$ for all i. i.e.,

$$A^n = \underbrace{A \times A \times \cdots \times A}_{n \text{ times}}$$

A *binary relation* $\mathcal{R}$ from A to B is a subset of $A \times B$. It is a characterization of the intuitive notion that some of the elements of A are related to some of the elements of B. We also use the notation $a\mathcal{R}b$ to mean $(a, b) \in \mathcal{R}$. When A and B are the same set, we say $\mathcal{R}$ is a binary relation *on* A. Familiar binary relations from $\mathbb{N}$ to $\mathbb{N}$ are $=$, $\neq$, $<$, $\leq$, $>$, $\geq$. Thus the elements of the set $\{(0,0), (0,1), (0,2), \ldots, (1,1), (1,2), \ldots\}$ are all members of the relation $\leq$ which is a subset of $\mathbb{N} \times \mathbb{N}$.

In general, an *n-ary relation* among the sets $A_1, A_2, \ldots, A_n$ is a subset of the set $A_1 \times A_2 \times \cdots \times A_n$.

Definition 1.1 *Let $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{B}$ be a binary relation from A to B. Then*

1. *For any set $A' \subseteq A$ the* image *of A' under $\mathcal{R}$ is the set defined by*

$$\mathcal{R}(A') = \{b \in B \mid a\mathcal{R}b \ for\ some\ a \in A'\}$$

2. *For every subset $B' \subseteq B$ the* pre-image *of B' under $\mathcal{R}$ is the set defined by*

$$\mathcal{R}^{-1}(B') = \{a \in A \mid a\mathcal{R}b \ for\ some\ b \in B'\}$$

3. *$\mathcal{R}$ is* onto *(or* surjective*) with respect to A and B if $\mathcal{R}(A) = B$.*

4. *$\mathcal{R}$ is* total *with respect to A and B if $\mathcal{R}^{-1}(B) = A$.*

5. *$\mathcal{R}$ is* one-to-one *(or* injective*) with respect to A and B if for every $b \in B$ there is at most one $a \in A$ such that $(a, b) \in \mathcal{R}$.*

6. *$\mathcal{R}$ is a* partial function *from A to B, usually denoted $\mathcal{R} : \mathcal{A} \hookrightarrow \mathcal{B}$, if for every $a \in A$ there is at most one $b \in B$ such that $(a, b) \in \mathcal{R}$.*

7. *$\mathcal{R}$ is a* total function *from A to B, usually denoted $\mathcal{R} : \mathcal{A} \longrightarrow \mathcal{B}$ if $\mathcal{R}$ is a partial function from A to B and is total.*

8. *$\mathcal{R}$ is a* one-to-one correspondence *(or* bijection*) if it is an injective and surjective total function.*

Notation. Let f be a function from set A to set B. Then

- $f : A \xrightarrow{\text{1-1}} B$ will denote that f is injective,

- $f : A \xrightarrow[\text{onto}]{} B$ will denote that f is surjective, and
- $f : A \xrightarrow[\text{onto}]{\text{1-1}} B$ will denote that f is bijective,

Example 1.1 *The following are some examples of familiar binary relations along with their properties.*

1. *The $\leq$ relation on $\mathbb{N}$ is a relation from $\mathbb{N}$ to $\mathbb{N}$ which is total and onto. That is, both the image and pre-image of $\leq$ under $\mathbb{N}$ are $\mathbb{N}$ itself. What are image and the pre-image respectively of the relation $<$?*
2. *The binary relation which associates key sequences from a computer keyboard with their respective 8-bit ASCII codes is an example of a relation which is total and injective.*
3. *The binary relation which associates 7-bit ASCII codes with the corresponding ASCII character set is an example of a bijection.*

The following figures illustrate the concepts of partial, injective, surjective, bijective and inverse of a bijective function on finite sets. The directed arrows go from elements in the domain to their images in the codomain.

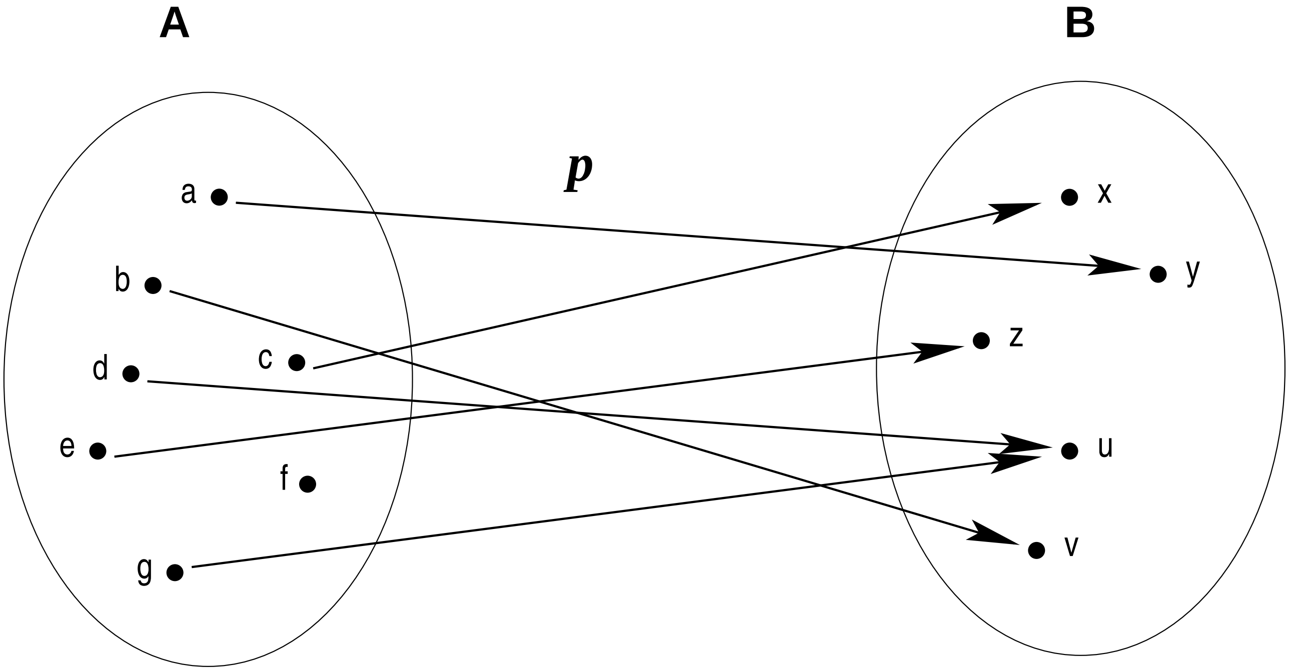


Figure 1.1: A partial function (*Why is it partial?*)

We may equivalently define partial and total functions as follows.

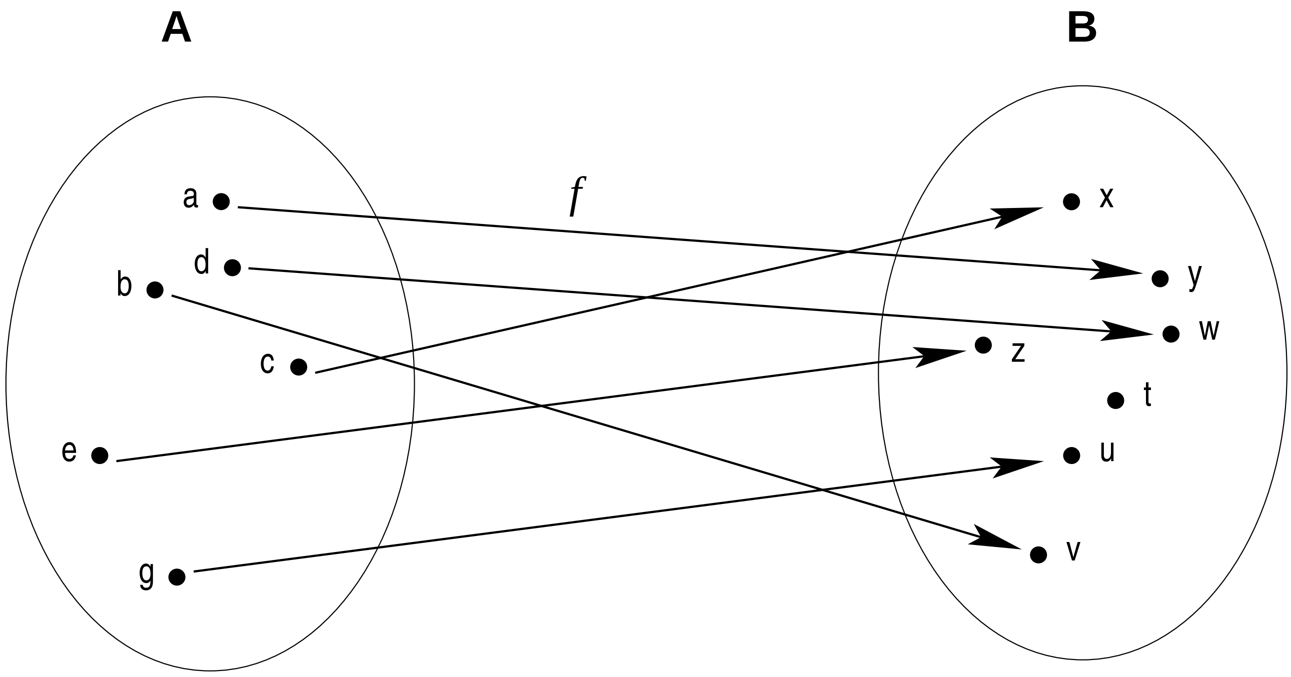


Figure 1.2: An injective function (*Why is it injective?*)

Definition 1.2 *A* function *(or a* total function*) f from A to B is a binary relation $f \subseteq A \times B$ such that for* every *element $a \in A$ there is a* unique *element $b \in B$ so that $(a, b) \in f$ (usually denoted $f(a) = b$ and sometimes $f : a \mapsto b$). We will use the notation $R : A \to B$ to denote a function R from A to B. The set A is called the* domain *of the function R and the set B is called the* co-domain *of the function R. The* range *of a function $R : A \to B$ is the set $\{b \in B \mid \text{for some } a \in A, R(a) = b\}$. A* partial function *$f$ from A to B, denoted $f : A \hookrightarrow B$ is a total function from some subset of A to the set B. Clearly every total function is also a partial function.*

The word "function" unless otherwise specified is taken to mean a "total function". Some familiar examples of partial and total functions are

1. $+$ and $*$ (addition and multiplication) are total functions of the type $f : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$
2. $-$ (subtraction) is a partial function of the type $f : \mathbb{N} \times \mathbb{N} \hookrightarrow \mathbb{N}$.
3. *div* and *mod* are total functions of the type $f : \mathbb{N} \times \mathbb{P} \to \mathbb{N}$. If $a = q * b + r$ such that $0 \leq r < b$ and $a, b, q, r \in \mathbb{N}$ then the functions *div* and *mod* are defined as $div(a, b) = q$ and $mod(a, b) = r$. We will often write these binary functions as $a * b$, $a\ div\ b$, $a\ mod\ b$ etc. Note that *div* and *mod* are also partial functions of the type $f : \mathbb{N} \times \mathbb{N} \hookrightarrow \mathbb{N}$.
4. The binary relations $=$, $\neq$, $<$, $\leq$, $>$, $\geq$ may also be thought of as functions of the type $f : \mathbb{N} \times \mathbb{N} \to \mathbb{B}$ where $\mathbb{B} = \{false, true\}$.

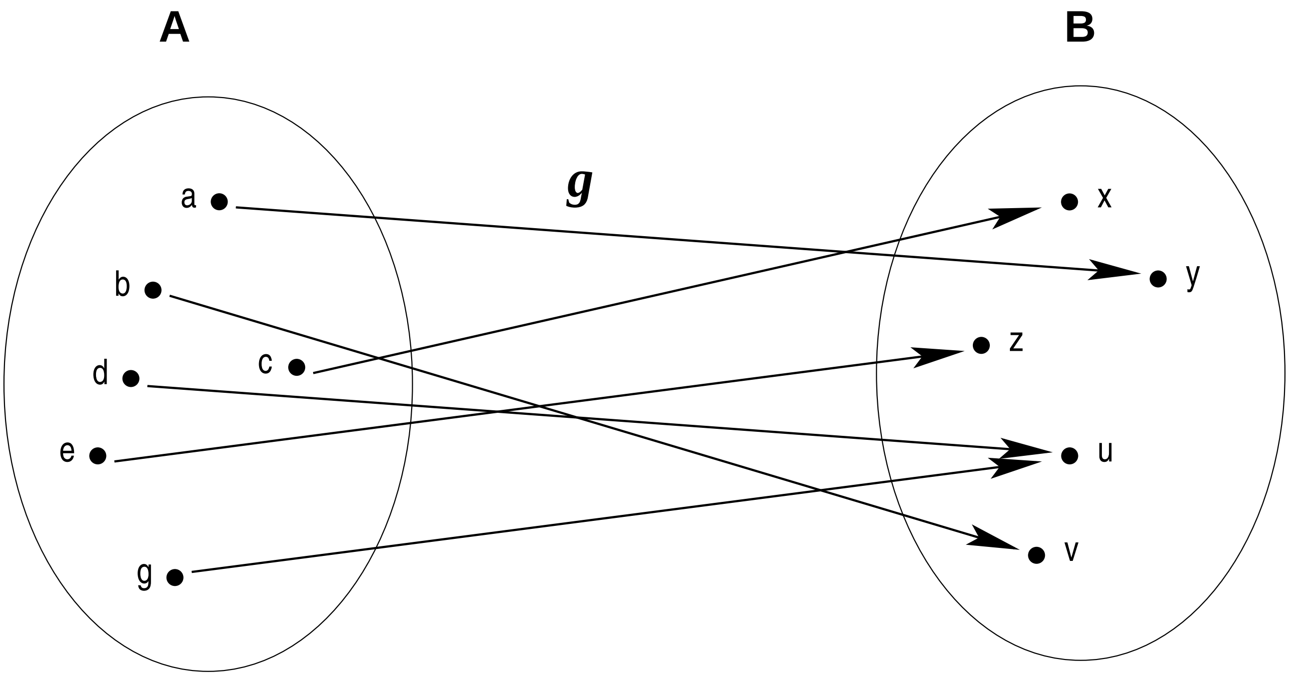


Figure 1.3: A surjective function (*Why is it surjective?*)

Definition 1.3 *Given a set A, a* list *(or* finite sequence*) of length $n \geq 0$ of elements from A, denoted $\vec{a}$, is a (total) function of the type $\vec{a} : \{1, 2, \ldots, n\} \rightarrow A$. We normally denote a list of length n by $[a_1, a_2, \ldots, a_n]$ Note that the empty list, denoted $[]$, is also such a function $[] : \emptyset \rightarrow A$ and denotes a sequence of length 0.*

It is quite clear that there exists a simple bijection from the set A^n (which is the set of all n-tuples of elements from the set A) and the set of all lists of length n of elements from A. We will often identify the two as being the same set even though they are actually different by definition[3]. The set of all lists of elements from A is denoted A^*, where

$$A^* = \bigcup_{n \geq 0} A^n$$

The set of all *non-empty* lists of elements from A is denoted A^+ and is defined as

$$A^+ = \bigcup_{n > 0} A^n$$

An *infinite* sequence of elements from A is a total function from $\mathbb{N}$ to A. The set of all such infinite sequences is denoted A^ω.

[3] In a programming language like ML, the difference is evident from the notation and the constructor operations for tuples and lists

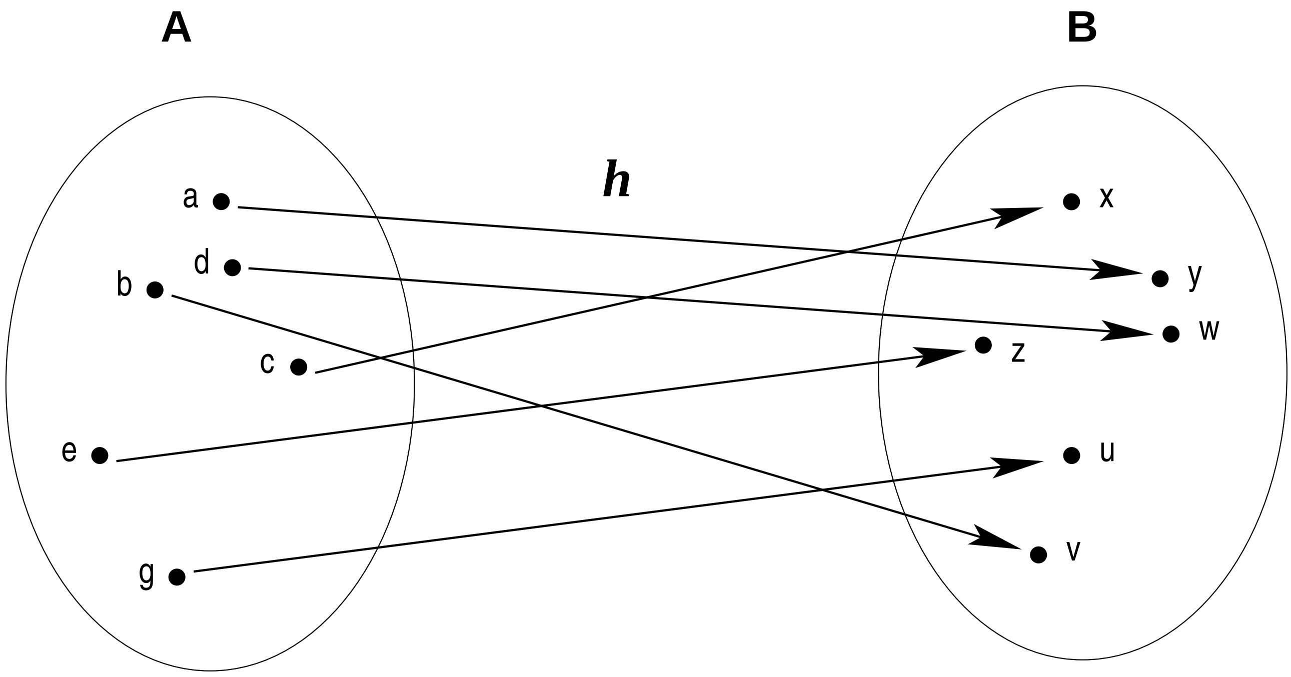


Figure 1.4: An bijective function (*Why is it bijective?*)

1.4 Operations on Binary Relations

In this section we will consider various operations on binary relations.

Definition 1.4 1. *Given a set A, the* identity *relation over A, denoted $\mathcal{I}_A$, is the set $\{\langle a, a\rangle \mid a \in A\}$.*

2. *Given a binary relation $\mathcal{R}$ from A to B, the* converse *of $\mathcal{R}$, denoted $\mathcal{R}^{-1}$ is the relation from B to A defined as $\mathcal{R}^{-1} = \{(b, a) \mid (a, b) \in \mathcal{R}\}$.*

3. *Given binary relations $\mathcal{R} \subseteq A \times B$ and $\mathcal{S} \subseteq B \times C$, the* composition *of $\mathcal{R}$ with $\mathcal{S}$ is denoted $\mathcal{R} \circ \mathcal{S}$ and defined as $\mathcal{R} \circ \mathcal{S} = \{(a, c) \mid a\mathcal{R}b \text{ and } b\mathcal{S}c, \ for \ some \ b \in B\}$.*

Note that unlike in the case of functions (where for any function $f : A \longrightarrow B$ its inverse $f^{-1} : B \longrightarrow A$ may not always be defined), the converse of a relation is always defined. Given functions (whether partial or total) $f : A \hookrightarrow B$ and $g : B \hookrightarrow C$, their composition is the function $f \circ g : A \hookrightarrow C$ defined simply as the relational composition of the two functions regarded as binary relations. Hence $(f \circ g)(a) = g(f(a))$.

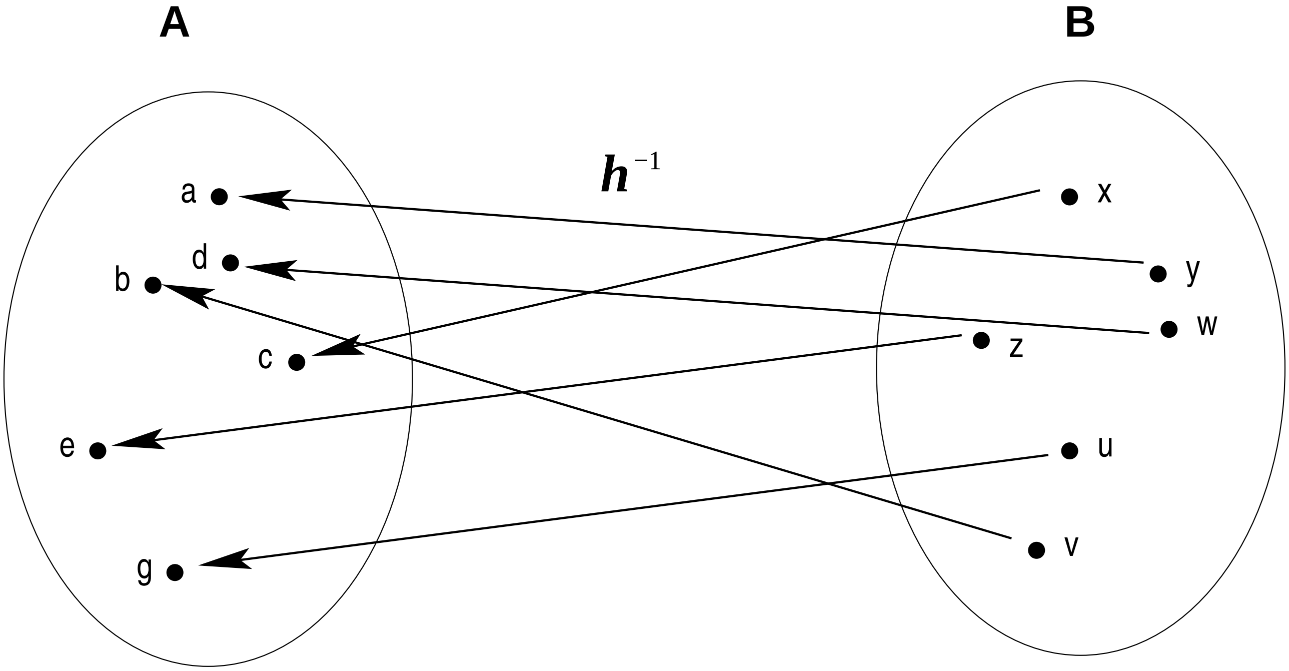


Figure 1.5: The inverse of the bijective function in Fig 1.4(*Is it bijective?*)

1.5 Ordering Relations

We may define the n-fold composition of a relation $\mathcal{R}$ on a set A by induction as follows

$$\mathcal{R}^0 = \mathcal{I}_A$$

$$\mathcal{R}^{n+1} = \mathcal{R}^n \circ R$$

We may combine these n-fold compositions to yield the *reflexive-transitive closure* of $\mathcal{R}$, denoted $\mathcal{R}$, as the relation

$$\mathcal{R}^* = \bigcup_{n \geq 0} \mathcal{R}^n$$

Sometimes it is also useful to consider merely the *transitive closure* $\mathcal{R}^+$ of $\mathcal{R}$ which is defined as

$$\mathcal{R}^+ = \bigcup_{n > 0} \mathcal{R}^n$$

Definition 1.5 *A binary relation $\mathcal{R}$ on a set A is*

1. reflexive *if and only if $\mathcal{I}_A \subseteq \mathcal{R}$;*
2. irreflexive *if and only if $\mathcal{I}_A \cap \mathcal{R} = \emptyset$;*
3. symmetric *if and only if $\mathcal{R} = \mathcal{R}^{-1}$;*

4. asymmetric *if and only if* $\mathcal{R} \cap \mathcal{R}^{-1} = \emptyset$;

5. antisymmetric *if and only if* $(a, b), (b, a) \in \mathcal{R}$ *implies* $a = b$.

6. transitive *if and only if for all* $a, b, c \in A$, $(a, b), (b, c) \in \mathcal{R}$ *implies* $(a, c) \in \mathcal{R}$.

7. connected *if and only if for all* $a, b \in A$, *if* $a \neq b$ *then* $a\mathcal{R}b$ *or* $b\mathcal{R}a$.

Given any relation $\mathcal{R}$ on a set A, it is easy to see that $\mathcal{R}^*$ is both reflexive and transitive.

Example 1.2 *1. The edge relation on an undirected graph is an example of a symmetric relation.*

2. *In any directed acyclic graph the edge relation is asymmetric.*

3. *Consider the reachability relation on a directed graph defined as: A pair of vertices* (A, B) *is in the reachability relation, if either* $A = B$ *or there exists a vertex* C *such that both* (A, C) *and* (C, B) *are in the reachability relation. The reachability relation is the reflexive transitive closure of the edge relation.*

4. *The reachability relation on directed graphs is also an example of a relation that need not be either symmetric or asymmetric. The relation need not be antisymmetric either.*

1.6 Partial Orders and Trees

Definition 1.6 *A binary relation* $\mathcal{R}$ *on a set* A *is*

1. *a* preorder *if it is reflexive and transitive;*

2. *a* strict preorder *if it is irreflexive and transitive;*

3. *a* partial order *if is an antisymmetric preorder;*

4. *a* strict partial order *if it is irreflexive, asymmetric and transitive;*

5. *a* linear order[4] *if it is a connected partial order;*

6. *a* strict linear order *if it is connected, irreflexive and transitive;*

7. *an* equivalence *if it is reflexive, symmetric and transitive.*

[4] also called *total order*

1.7 Infinite Sets: Countability and Uncountability

Definition 1.7 *A set A is finite if it can be placed in bijection with a set $\{m \in \mathbb{P} | m < n\}$ for some $n \in \mathbb{N}$.*

The above definition embodies the usual notion of counting. Since it is intuitively clear we shall not have anything more to say about.

Definition 1.8 *A set A is called* **infinite** *if there exists a bijection between A and some proper subset of itself.*

This definition begs the question, "If a set is not infinite, then is it necessarily finite?". It turns out that indeed it is. Further it is also true that if a set is not finite then it can be placed in $1-1$-correspondence with a proper subset of itself. But the proofs of these statements are beyond the scope of this chapter and hence we shall not pursue them.

Example 1.3 *We give appropriate 1-1 correspondences to show that various sets are infinite. In each case, note that the codomain of the bijection is a proper subset of the domain.*

1. *The set $\mathbb{N}$ of natural numbers is infinite because we can define the 1-1 correspondence $p : \mathbb{N} \xrightarrow[onto]{1\text{-}1} \mathbb{P}$, with $p(m) \triangleq m + 1$.*
2. *The set E of even natural numbers is infinite because we have the bijection $e : E \xrightarrow[onto]{1\text{-}1} F$ where F is the set of all multiples of 4.*
3. *The set of odd natural numbers is infinite. (Why?)*
4. *The set $\mathbb{Z}$ of integers is infinite because we have the following bijection $z : \mathbb{Z} \xrightarrow[onto]{1\text{-}1} \mathbb{N}$ by which the negative integers have unique images among the odd numbers and the non-negative integers have unique images among the even numbers. More specifically,*

$$z(m) = \begin{cases} 2m & \text{if } m \in \mathbb{N} \\ -2m-1 & \text{otherwise} \end{cases}$$

Example 1.4 *The set $\mathbb{R}$ of reals is infinite. To prove this let us consider the open interval (a, b) and use figure 1.6 as a guide to understand the mapping.*

Take any line-segment $\overline{AB}$ of length $b - a \neq 0$ and bend it into the semi-circle $\overset{\frown}{A'B'}$ and place it tangent to the x-axis at the point $(0, 0)$ (as shown in the figure). This semicircle has a radius $r = \dfrac{b-a}{\pi}$. The centre C of this semi-circle is then located at the point $(0, r)$ on the 2-dimensional plane.

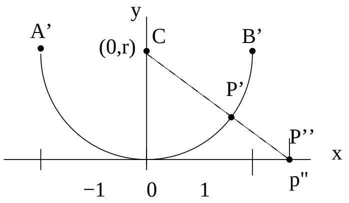


Figure 1.6: Bijection between the arc $A'B'$ and the real line

Each point P' such that $A' \neq P' \neq B'$ on this semi-circle corresponds exactly to a unique real number p in the open interval (a, b) and vice-versa. Further the ray $\overrightarrow{CP'}$ always intesects the x-axis at some point P''. There exists a 1-1 correspondence between each such P' and P'' on the x-axis. Let p'' be the x-coordinate of the point P''. Since the composition of bijections is a bijection, we may compose all these bijections to obtain a 1-1 correspondence between each p in the interval (a, b) and the real numbers.

Definition 1.9 *A set is said to be* **countable** *(or* **countably infinite***) if it can be placed in bijection with the set of natural numbers. Otherwise, it is said to be* **uncountable***.*

Fact 1.1 *The following are easy to prove.*

1. *Every infinite subset of $\mathbb{N}$ is countable.*
2. *If A is a finite set and B is a countable set, then $A \cup B$ is countable.*
3. *If A and B are countable sets, then $A \cup B$ is also countable.*

Theorem 1.2 $\mathbb{N}^2$ *is a countable set.*

Proof:

We show that $\mathbb{N}^2$ is countably infinite by devising a way to order the elements of $\mathbb{N}^2$ which guarantees that there is indeed a 1-1 correspondence. For instance, an obvious ordering such as

$$\begin{array}{ccccc}
(0,0) & (0,1) & (0,2) & (0,3) & \dots \\
(1,0) & (1,1) & (1,2) & (1,3) & \dots \\
(2,0) & (2,1) & (2,2) & (2,3) & \dots \\
\vdots & & \ddots & & \dots
\end{array}$$

is not a 1-1 correspondence because we cannot answer the following questions with (unique) answers.

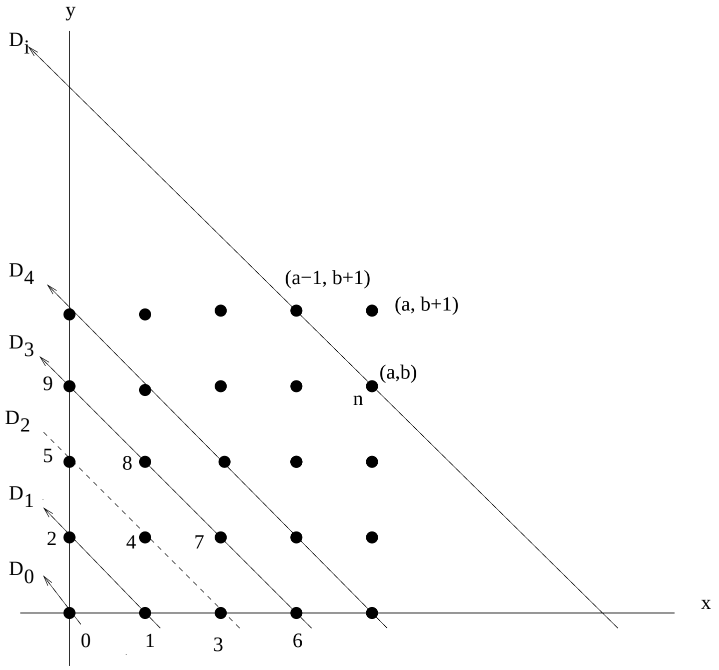


Figure 1.7: Counting "lattice-points" on the "diagonals"

1. *What is the n-th element in the ordering?*
2. *What is the position in the ordering of the pair (a, b)?*

So it is necessary to construct a more rigorous and ingenious device to ensure a bijection. So we consider the ordering implicitly defined in figure 1.7. By traversing the rays $\overrightarrow{D_0}$, $\overrightarrow{D_1}$, $\overrightarrow{D_2}$, ...in order, we get an obvious ordering on the elements of $\mathbb{N}^2$. However it should be possible to give unique answers to the above questions.

Claim $f : \mathbb{N}^2 \longrightarrow \mathbb{N}$ defined by $f(a, b) = \dfrac{(a+b)(a+b+1)+2b}{2}$ is the required bijection.

Proof outline: The function f defines essentially the traversal of the rays $\overrightarrow{D_0}$, $\overrightarrow{D_1}$, $\overrightarrow{D_2}$, ...in order as we shall prove. It is easy to verify that $\overrightarrow{D_0}$ contains only the pair $(0, 0)$ and $f(0, 0) = 0$. Now consider any pair $(a, b) \neq (0, 0)$. If (a, b) lies on the ray $\overrightarrow{D_i}$, then it is clear that $i = a + b$. Now consider all the pairs that lie on the rays $\overrightarrow{D_0}$, $\overrightarrow{D_1}$, ..., $\overrightarrow{D_{i-1}}$[5]

The number of such pairs is given by the "triangular number"

$$i + (i-1) + (i-2) + \cdots + 1 = \frac{i(i+1)}{2}$$

[5] Under the usual (x, y) coordinate system, these are all the *lattice points* on and inside the right triangle defined by the three points $(i-1, 0)$, $(0, 0)$ and $(0, i-1)$. A *lattice point* in the (x, y)-plane is point whose $x-$ and $y-$ coordinates are both integers.

Since we started counting from 0 this number is also the value of the lattice point $(i, 0)$ under the function f. This brings us to the starting point of the ray D_i and after crossing b lattice points along the ray D_i we arrive at the point (a, b). Hence

$$\begin{aligned} f(a,b) &= \frac{i(i+1)}{2} + b \\ &= \frac{(a+b)(a+b+1)+2b}{2} \end{aligned}$$

We leave it as an exercise to the reader to define the inverse of this function. (*Hint: Use "triangular numbers"!*) □ □

Example 1.5 *Let the language $\mathcal{M}_0$ of minimal logic be "generated" by the following process from a countably infinite set of "atoms" $\mathbb{A}$, such that $\mathbb{A}$ does not contain any of the symbols "¬", "→", "(" and ")".*

1. $\mathbb{A} \subseteq \mathcal{M}_0$,

2. *If μ and ν are any two elements of $\mathcal{M}_0$ then $(\neg\mu)$ and $(\mu \rightarrow \nu)$ also belong to $\mathcal{M}_0$, and*

3. *No string other than those obtained by a finite number of applications of the above rules belongs to $\mathcal{M}_0$.*

set. We prove that the $\mathcal{M}_0$ is countably infinite.

Solution *There are at least two possible proofs. The first one simply encodes of formulas into unique natural numbers. The second uses induction on the structure of formulas and the fact that a countable union of countable sets yields a countable set. We postpone the second proof to the chapter on induction. So here goes!*

Proof: *Since $\mathbb{A}$ is countably infinite, there exists a $1-1$ correspondence $ord : \mathbb{A} \leftrightarrow \mathbb{P}$ which uniquely enumerates the atoms in some order. This function may be extended to a function ord' which includes the symbols "¬", "(", ")", "→", such that $ord'(``\neg") = 1$, $ord'(``(") = 2$, $ord'(``)") = 3$, $ord'(``\rightarrow") = 4$, and $ord'(``A") = ord(``A") + 4$, for every $A \in \mathbb{A}$. Let $\mathsf{Syms} = \mathbb{A} \cup \{``\neg", ``(", ``)", ``\rightarrow"\}$. Clearly $ord' : \mathsf{Syms} \leftrightarrow \mathbb{P}$ is also a $1-1$ correspondence. Hence there also exist inverse functions ord^{-1} and ord'^{-1} which for any positive integer identify a unique symbol from the domains of the two functions respectively.*

Now consider any string[6] belonging to Syms^*. It is possible to assign a unique positive integer to this string by using powers of primes. Let $p_1 = 2$, $p_2 = 3, \ldots, p_i, \ldots$ be the infinite list of primes in increasing order. Let the function $encode : \mathsf{Syms}^* \longrightarrow \mathbb{P}$ be defined by induction on the lengths of the strings in Syms^*, as follows. Assume $s \in \mathsf{Syms}^*$, $a \in \mathsf{Syms}$ and "" denotes the empty string.

$$\begin{aligned} encode(``") &= 1 \\ encode(sa) &= encode(s) \times {p_m}^{ord'(a)} \end{aligned}$$

[6] This includes even arbitrary strings which are not part of the language. For example, you may have strings such as ")¬(".

where s is a string of length $m-1$ for $m \geq 1$.

It is now obvious from the unique prime-factorization of positive integers that every string in Syms^* *has a unique positive integer as its "encoding" and from any positive integer it is possible to get the unique string that it represents. Hence* Syms^* *is a countably infinite set. Since the language of minimal logic is a subset of the* Syms^* *it cannot be an* uncountably *infinite set. Hence there are only two possibilities: either it is finite or it is countably infinite.*

Claim. The language of minimal logic is not finite.
Proof of claim. Suppose the language were finite. Then there exists a formula ϕ in the language such that encode(ϕ) is the maximum possible positive integer. This $\phi \in \mathsf{Syms}^*$ and hence is a string of the form $a_1 \ldots a_m$ where each $a_i \in \mathsf{Syms}$. Clearly

$$encode(\phi) = \prod_{i=1}^{m} {p_i}^{ord'(a_i)}$$

. Now consider the longer formula $\psi = (\neg\phi)$. It is easy to show that

$$encode(\psi) = 2^{ord'(``(``)} \times 3^{ord'(``\neg'')} \times \prod_{i=1}^{m} {p_{i+2}}^{ord'(a_i)} \times {p_{m+3}}^{ord'(``)'')}$$

and encode(ψ) > encode(ϕ) contradicting the assumption of the claim.

Hence the language is countably infinite. □

Not all infinite sets that can be constructed are countable. In other words even among infinite sets there are some sets that are "more infinite than others". The following theorem and the form of its proof was first given by Georg Cantor and has been used to prove several results in logic, mathematics and computer science.

Theorem 1.3 (Cantor's diagonalization). *The powerset of $\mathbb{N}$ (i.e. $\mathbf{2}^{\mathbb{N}}$, the set of all subsets of $\mathbb{N}$) is an uncountable set.*

Proof: Firstly, it should be clear that $\mathbf{2}^{\mathbb{N}}$ is not a finite set, since for every natural number n, the singleton set $\{n\}$ belongs to $\mathbf{2}^{\mathbb{N}}$.

Consider any subset $A \subseteq \mathbb{N}$. We may represent this set as an infinite sequence σ_A composed of 0's and 1's such that $\sigma_A(i) = 1$ if $i \in A$, otherwise $\sigma_A(i) = 0$. Let $\Sigma = \{\sigma | \forall i \in \mathbb{N} : \sigma(i) \in \{0, 1\}\}$ be the set of all such sequences. It is easy to show that there exists a bijection $g : \mathbf{2}^{\mathbb{N}} \xrightarrow[\text{onto}]{\text{1-1}} \Sigma$ such that $g(A) = \sigma_A$, for each $A \subseteq \mathbb{N}$. Clearly, therefore $\mathbf{2}^{\mathbb{N}}$ is countable if and only if Σ is countable. Hence, if there exists a bijection $f : \Sigma \xrightarrow[\text{onto}]{\text{1-1}} \mathbb{N}$, then $f \circ g$ is the required bijection from $\mathbf{2}^{\mathbb{N}}$ to $\mathbb{N}$. On the other hand, if there is no bijection f then $\mathbf{2}^{\mathbb{N}}$ is uncountable if and only if Σ is uncountable. We make the following claim which we prove by Cantor's diagonalization.

Claim 1.1 *The set Σ is uncountable.*

We prove the claim as follows. Suppose Σ is countable then there exists a bijection $h : \mathbb{N} \xrightarrow[\text{onto}]{\text{1-1}} \Sigma$. In fact let $h(i) = \sigma_i \in \Sigma$, for each $i \in \mathbb{N}$. Now consider the sequence ρ constructed in such a manner that for each $i \in \mathbb{N}$, $\rho(i) \neq \sigma_i(i)$. In other words,

$$\rho(i) = \begin{cases} 0 & \text{if } \sigma_i(i) = 1 \\ 1 & \text{if } \sigma_i(i) = 0 \end{cases}$$

Since ρ is an infinite sequence of 0's and 1's, $\rho \in \Sigma$. But from the above construction it follows that since ρ is different from every sequence in Σ it cannot be a member of Σ, leading to a contradiction. Hence the assumption that Σ is uncountable must be wrong. □

1.8 Exercises

1. Prove that for any binary relations $\mathcal{R}$ and $\mathcal{S}$ on a set A,

   (a) $(\mathcal{R}^{-1})^{-1} = \mathcal{R}$

   (b) $(\mathcal{R} \cap \mathcal{S})^{-1} = \mathcal{R}^{-1} \cap \mathcal{S}^{-1}$

   (c) $(\mathcal{R} \cup \mathcal{S})^{-1} = \mathcal{R}^{-1} \cup \mathcal{S}^{-1}$

   (d) $(\mathcal{R} - \mathcal{S})^{-1} = \mathcal{R}^{-1} - \mathcal{S}^{-1}$

2. Prove that the composition operation on relations is associative. Give an example of the composition of relations to show that relational composition is not commutative.

3. Prove that for any binary relations $\mathcal{R}$, $\mathcal{R}'$ from A to B and $\mathcal{S}$, $\mathcal{S}'$ from B to C, if $\mathcal{R} \subseteq \mathcal{R}'$ and $\mathcal{S} \subseteq \mathcal{S}'$ then $\mathcal{R} \circ \mathcal{S} \subseteq \mathcal{R}' \circ \mathcal{S}'$

4. Prove or disprove[7]that relational composition satisfies the following distributive laws for relations, where $\mathcal{R} \subseteq A \times B$ and $\mathcal{S}, \mathcal{T} \subseteq B \times C$.

   (a) $\mathcal{R} \circ (\mathcal{S} \cup \mathcal{T}) = (\mathcal{R} \circ \mathcal{S}) \cup (\mathcal{R} \circ \mathcal{T})$

   (b) $\mathcal{R} \circ (\mathcal{S} \cap \mathcal{T}) = (\mathcal{R} \circ \mathcal{S}) \cap (\mathcal{R} \circ \mathcal{T})$

   (c) $\mathcal{R} \circ (\mathcal{S} - \mathcal{T}) = (\mathcal{R} \circ \mathcal{S}) - (\mathcal{R} \circ \mathcal{T})$

5. Prove that for $\mathcal{R} \subseteq A \times B$ and $\mathcal{S} \subseteq B \times C$, $(\mathcal{R} \circ \mathcal{S})^{-1} = (\mathcal{S}^{-1}) \circ (\mathcal{R}^{-1})$.

6. Show that a relation $\mathcal{R}$ on a set A is

   (a) antisymmetric if and only if $\mathcal{R} \cap \mathcal{R}^{-1} \subseteq \mathcal{I}_A$

   (b) transitive if and only if $\mathcal{R} \circ \mathcal{R} \subseteq \mathcal{R}$

[7] that is, find an example of appropriate relations which actually violate the equality

   (c) connected if and only if $(A \times A) - \mathcal{I}_A \subseteq \mathcal{R} \cup \mathcal{R}^{-1}$

7. Consider any reflexive relation $\mathcal{R}$ on a set A. Does it necessarily follow that A is not asymmetric? If $\mathcal{R}$ is asymmetric does it necessarily follow that it is irreflexive?

8. Prove that

   (a) $\mathbb{N}^n$, for any $n > 0$ is a countably infinite set,

   (b) If $\{A_i | i \geq 0\}$ is a countable collection of pair-wise disjoint sets (i.e. $A_i \cap A_j = \emptyset$ for all $i \neq j$) then $A = \bigcup_{i \geq 0} A_i$ is also a countable set.

   (c) $\mathbb{N}^*$ the set of all finite sequences of natural numbers is countable.

9. Prove that

   (a) $\mathbb{N}^\omega$ the set of all *infinite* sequences of natural numbers is uncountable,

   (b) the set of all binary relations on a countably infinite set is an uncountable set,

   (c) the set of all total functions from $\mathbb{N}$ to $\mathbb{N}$ is uncountable.

10. Prove that there exists a bijection between the set $\mathbf{2}^{\mathbb{N}}$ and the open interval $(0, 1)$ of real numbers. *Question: How do you handle numbers that are equal but have 2 different decimal representations such as* $0.8\bar{9}$ *and* 0.9*?*. What can you conclude about the cardinality of the set $\mathbf{2}^{\mathbb{N}}$ in relation to the set $\mathbb{R}$?

11. Prove that for any binary relations $\mathcal{R}$ and $\mathcal{S}$ on a set A,

    (a) $(\mathcal{R}^{-1})^{-1} = \mathcal{R}$

    (b) $(\mathcal{R} \cap \mathcal{S})^{-1} = \mathcal{R}^{-1} \cap \mathcal{S}^{-1}$

    (c) $(\mathcal{R} \cup \mathcal{S})^{-1} = \mathcal{R}^{-1} \cup \mathcal{S}^{-1}$

    (d) $(\mathcal{R} - \mathcal{S})^{-1} = \mathcal{R}^{-1} - \mathcal{S}^{-1}$

12. Prove that the composition operation on relations is associative. Give an example of the composition of relations to show that relational composition is not commutative.

13. Prove that for any binary relations $\mathcal{R}$, $\mathcal{R}'$ from A to B and $\mathcal{S}$, $\mathcal{S}'$ from B to C, if $\mathcal{R} \subseteq \mathcal{R}'$ and $\mathcal{S} \subseteq \mathcal{S}'$ then $\mathcal{R} \circ \mathcal{S} \subseteq \mathcal{R}' \circ \mathcal{S}'$

14. Prove or disprove[8]that relational composition satisfies the following distributive laws for relations, where $\mathcal{R} \subseteq A \times B$ and $\mathcal{S}, \mathcal{T} \subseteq B \times C$.

    (a) $\mathcal{R} \circ (\mathcal{S} \cup \mathcal{T}) = (\mathcal{R} \circ \mathcal{S}) \cup (\mathcal{R} \circ \mathcal{T})$

    (b) $\mathcal{R} \circ (\mathcal{S} \cap \mathcal{T}) = (\mathcal{R} \circ \mathcal{S}) \cap (\mathcal{R} \circ \mathcal{T})$

    (c) $\mathcal{R} \circ (\mathcal{S} - \mathcal{T}) = (\mathcal{R} \circ \mathcal{S}) - (\mathcal{R} \circ \mathcal{T})$

15. Prove that for $\mathcal{R} \subseteq A \times B$ and $\mathcal{S} \subseteq B \times C$, $(\mathcal{R} \circ \mathcal{S})^{-1} = (\mathcal{S}^{-1}) \circ (\mathcal{R}^{-1})$.

[8] that is, find an example of appropriate relations which actually violate the equality

16. Show that a relation $\mathcal{R}$ on a set A is

    (a) antisymmetric if and only if $\mathcal{R} \cap \mathcal{R}^{-1} \subseteq \mathcal{I}_A$

    (b) transitive if and only if $\mathcal{R} \circ \mathcal{R} \subseteq \mathcal{R}$

    (c) connected if and only if $(A \times A) - \mathcal{I}_A \subseteq \mathcal{R} \cup \mathcal{R}^{-1}$

17. Consider any reflexive relation $\mathcal{R}$ on a set A. Does it necessarily follow that $\mathcal{R}$ is not asymmetric? If $\mathcal{R}$ is asymmetric does it necessarily follow that it is irreflexive?

18. Prove that for any relation $\mathcal{R}$ on a set A,

    (a) $\mathcal{S} = \mathcal{R}^* \cup (\mathcal{R}^*)^{-1}$ and $\mathcal{T} = (\mathcal{R} \cup \mathcal{R}^{-1})^*$ are both equivalence relations.

    (b) Prove or disprove: $\mathcal{S} = \mathcal{T}$.

19. Given any preorder $\mathcal{R}$ on a set A, prove that the *kernel* of the preorder defined as $\mathcal{R} \cap \mathcal{R}^{-1}$ is an equivalence relation.

20. Consider any preorder $\mathcal{R}$ on a set A. We give a construction of another relation as follows. For each $a \in A$, let $[a]_\mathcal{R}$ be the set defined as $[a]_\mathcal{R} = \{b \in A \mid a\mathcal{R}b \textit{ and } b\mathcal{R}a\}$. Now consider the set $B = \{[a]_\mathcal{R} \mid a \in A\}$. Let $\mathcal{S}$ be a relation on B such that for every $a, b \in A$, $[a]_\mathcal{R}\mathcal{S}[b]_\mathcal{R}$ if and only if $a\mathcal{R}b$. Prove that $\mathcal{S}$ is a partial order on the set B.

Chapter 2

Induction Principles

> **Theorem***: All natural numbers are equal.*
> *Proof: Given a pair of natural numbers a and b, we prove they are equal by performing complete induction on the maximum of a and b (denoted $max(a, b)$).*
> Basis. *For all natural numbers less than or equal to 0, the claim holds.*
> Induction hypothesis. *For any a and b such that $max(a, b) \leq k$, for some natural $k \geq 0$, $a = b$.*
> Induction step. *Let a and b be naturals such that $max(a, b) = k + 1$. It follows that $max(a - 1, b - 1) = k$. By the induction hypothesis $a - 1 = b - 1$. Adding 1 on both sides we get $a = b$ QED.*
>
> Fortune cookie on Linux

2.1 Mathematical Induction

Anyone who has had a good background in school mathematics must be familiar with two uses of induction.

1. definition of functions and relations by mathematical induction, and
2. proofs by the principle of mathematical induction.

Example 2.1 *We present below some familiar examples of definitions by mathematical induction.*

1. *The factorial function on natural numbers is defined as follows.*

   Basis. $0! = 1$

Induction step. $(n+1)! = n! \times (n+1)$

2. *The n-th power (where n is a natural number) of a real number x is often defined as*

   Basis. $x^1 \;=\; x$

   Induction step. $x^{n+1} \;=\; x^n \;\times\; x$

3. *For binary relations R, S on A we define their composition (denoted $R \circ S$) as follows.*

$$R \circ S = \{(a, c) \mid \textit{ for some } b \in A, (a, b) \in R \textit{ and } (b, c) \in S\}$$

   We may extend this binary relational composition to an n-fold composition of a single relation R as follows.

   Basis. $R^1 = R$

   Induction step. $R^{n+1} = R \circ R^n$

Similarly the principle of mathematical induction is the means by which we have often *proved* (as opposed to *defining*) properties about numbers, or statements involving the natural numbers. The principle may be stated as follows.

Principle of Mathematical Induction – Version 1

A property **P** *holds for all natural numbers provided*

Basis. **P** *holds for* 0*, and*

Induction step. *For arbitrarily chosen* $n > 0$*,*
$\underline{P \textit{ holds for } n-1}$ *implies P holds for n.*

The underlined portion, called the **induction hypothesis**, is an assumption that is necessary for the conclusion to be proved. Intuitively, the principle captures the fact that in order to prove any statement involving natural numbers, it is suffices to do it in two steps. The first step is the basis and needs to be proved. The proof of the induction step essentially tells us that the reasoning involved in proving the statement for all other natural numbers is the same. Hence instead of an infinitary proof (one for each natural number) we have a compact finitary proof which exploits the similarity of the proofs for all the naturals except the basis.

Example 2.2 *We prove that all natural numbers of the form n^3+2n are divisible by* 3*. Proof:*

Basis. *For $n = 0$, we have $n^3 + 2n = 0$ which is divisible by* 3*.*

Induction step. *Assume for an arbitrarily chosen* $n \geq 0$, $n^3 + 2n$ *is divisible by* 3. *Now consider* $(n+1)^3 + 2(n+1)$. *We have*

$$\begin{array}{rcl} (n+1)^3 + 2(n+1) & = & (n^3 + 3n^2 + 3n + 1) + (2n + 2) \\ & = & (n^3 + 2n) + 3(n^2 + n + 1) \end{array}$$

which clearly is divisible by 3.

□

Several versions of this principle exist. We state some of the most important ones. In such cases, the underlined portion is the induction hypothesis. For example it is not necessary to consider 0 (or even 1) as the basis step. Any integer k could be considered the basis, as long as the property is to be proved for all $n \geq k$.

Principle of Mathematical Induction – Version 2

A property **P** *holds for all natural numbers* $n \geq k$ *for some natural number* k, *provided*

Basis. P *holds for* k, *and*

Induction step. *For arbitrarily chosen* $n > k$,
$\underline{P \text{ holds for } n-1}$ *implies* P *holds for* n.

Such a version seems very useful when the property to be proved is either not true or is undefined for all naturals less than k. The following example illustrates this.

Example 2.3 *Every positive integer* $n \geq 8$ *is expressible as* $n = 3i + 5j$ *where* $i, j \geq 0$.
Proof: .

Basis. *For* $n = 8$, *we have* $n = 3 + 5$, *i.e.* $i = j = 1$.

Induction step. *Assuming for an arbitrary* $n \geq 8$, $n = 3i + 5j$ *for some naturals* i *and* j, *consider* $n + 1$. *If* $j = 0$ *then clearly* $i \geq 3$ *and we may write* $n + 1$ *as* $3(i - 3) + 5(j + 2)$. *Otherwise* $n + 1 = 3(i + 2) + 5(j - 1)$.

□

However it is not necessary to have this new version of the Principle of mathematical induction at all as the following reworking of the previous example shows.

Example 2.4 *The property of the previous example could be equivalently reworded as follows.* "For every natural number n, $n+8$ is expressible as $n+8 = 3i+5j$ where $i, j \geq 0$".
Proof: .

Basis. *For $n = 0$, we have $n+8 = 8 = 3+5$, i.e. $i = j = 1$.*

Induction step. *Assuming for an arbitrary $n \geq 0$, $n+8 = 3i+5j$ for some naturals i and j, consider $n+1$. If $j = 0$ then clearly $i \geq 3$ and we may write $(n+1)+8$ as $3(i-3)+5(j+2)$. Otherwise $(n+1)+8 = 3(i+2)+5(j-1)$.*

□

In general any property **P** that holds for all naturals greater than or equal to some given k may be transformed equivalently into a property **Q**, which reads exactly like **P** except that all occurrences of "n" in **P** are systematically replaced by "$n+k$". We may then prove the property **Q** using the first version of the principle.

What we have stated above informally is, in fact a proof outline of the following theorem.

Theorem 2.1 *The two principles of mathematical induction are equivalent. In other words, every application of PMI - version 1 may be transformed into an application of PMI – version 2 and vice-versa.*

In the sequel we will assume that the principle of mathematical induction always refers to the first version.

2.2 Complete Induction

Often in inductive definitions and proofs it seems necessary to work an inductive hypothesis that includes not just the predecessor of a natural number, but some or all of their predecessors as well.

Example 2.5 *The definition of the following sequence is a case of precisely such a definition where the function $F(n)$ is defined for all naturals as follows.*

Basis. $F(0) = 0$

Induction step

$$F(n+1) = \begin{cases} 1 & \textit{if } n = 0 \\ F(n)+F(n-1) & \textit{otherwise} \end{cases}$$

This is the famous Fibonacci[1] *sequence.*

One of the properties of the Fibonacci sequence is that the sequence converges to the "golden ratio"[2]. For any inductive proof of properties of the Fibonacci numbers, we would clearly need to assume that the property holds for the two preceding numbers in the sequence.

In the following, we present a principle that assumes a stronger induction hypothesis.

Principle of Complete Induction (PCI)

A property **P** *holds for all natural numbers provided*

Basis. **P** *holds for* 0.

Induction step. *For an arbirary* $n > 0$

$\underline{\mathbf{P} \textit{ holds for every } m,\ 0 \leq m < n}$ *implies* **P** *holds for* n

---

Example 2.6 *Let* $F(0) = 0,\ F(1) = 1,\ F(2) = 1,\ \ldots,\ F(n+1) = F(n) + F(n-1),\ \ldots$ *be the Fibonacci sequence. Let* ϕ *be the "golden ratio"* $(1 + 5\sqrt{5})/2$. *We now show that the property* $F(n+1) \leq \phi^n$ *holds for all* n.
Proof: *By the principle of complete induction on* n.

Basis. *For* $n = 0$, *we have* $F(1) = \phi^0 = 1$.

Induction step. *Assuming the property holds for all* m, $0 \leq m \leq n-1$, *for an arbitrarily chosen* $n > 0$, *we need to prove that* $F(n+1) \leq \phi^n$.

$$\begin{array}{rcll} F(n+1) & = & F(n) + F(n-1) & \\ & \leq & \phi^{n-1} + \phi^{n-2} & \textit{by the induction hypothesis} \\ & = & \phi^{n-2}(\phi + 1) & \\ & = & \phi^n & \textit{since } \phi^2 = \phi + 1 \end{array}$$

□

Note that the feature distinguishing the principle of mathematical induction from that of complete induction is the induction hypothesis. It appears to be much stronger in the latter case. However, in the following example we again prove the property in example 2.6 but this time we use the principle of mathematical induction instead.

[1] named after Leonardo of Fibonacci.

[2] one of the solutions of the equation $x^2 = x + 1$. It was considered an aesthetically pleasing aspect ratio for buildings in ancient Greek architecture.

Example 2.7 *Let $\mathbf{P}(n)$ denote the property*

$$\text{“}F(n+1) \leq \phi^n.\text{”}$$

*Rather than prove the original statement “*For all *n, $\mathbf{P}(n)$” we instead consider the property $\mathbf{Q}(n)$ which we define as*

$$\text{“For every } m,\ 0 \leq m \leq n,\ \mathbf{P}(m).\text{”}$$

*and prove the statement “*For all *n, $\mathbf{Q}(n)$”. This property can now be proved by mathematical induction as follows. The reader is encouraged to study the following proof carefully.*
Proof: *By the principle of mathematical induction on n.*

Basis. *For $n = 0$, we have $F(1) = \phi^0 = 1$.*

Induction step. *Assuming the property $\mathbf{Q}(n-1)$, holds for an arbitrarily chosen $n > 0$, we need to prove the property $\mathbf{Q}$ for n. But for this it suffices to prove the property $\mathbf{P}$ for n, since $\mathbf{Q}(n)$ is equivalent to the conjunction of $\mathbf{Q}(n-1)$ and $\mathbf{P}(n)$. Hence we prove the property $\mathbf{P}(n)$.*

$$\begin{array}{rcll}
F(n+1) & = & F(n) + F(n-1) & \\
& \leq & \phi^{n-1} + \phi^{n-2} & \textit{by the induction hypothesis} \\
& = & \phi^{n-2}(\phi + 1) & \\
& = & \phi^n & \textit{since } \phi^2 = \phi + 1
\end{array}$$

□

The above example shows quite clearly that the induction hypothesis used in any application of complete induction though seemingly stronger, can also lead to the proof of seemingly stronger properties. But in fact, in the end the proofs are almost identical. These proofs lead us then naturally into the next theorem.

Theorem 2.2 *The two principles of mathematical induction are equivalent. In other words, every application of PMI may be transformed into an application of PCI and vice-versa.*

Proof: We need to prove the following two claims.

1. *Any proof of a property using the principle of mathematical induction, is also a proof of the same property using the principle of complete induction.* This is so because the only possible change in the nature of two proofs could be because they use different induction hypotheses. Since the proof by mathematical induction uses a fairly weak assumption which is sufficient to prove the property, strengthening it in any way does not need to change the rest of the proof of the induction step.

2. *For every proof of a property using the principle of complete induction, there exists a corresponding proof of the same property using the principle of mathematical induction.* To prove this claim we resort to the same trick employed in example 2.6. We merely replace each occurrence of the original property in the form $\mathbf{P}(n)$ by $\mathbf{Q}(n)$, where the property $\mathbf{Q}$ is defined as

   "For every m, $0 \leq m \leq n$, $\mathbf{P}(m)$."

   Since $\mathbf{Q}(0)$ is the same as $\mathbf{P}(0)$ there is no other change in the basis step of the proof. In the original proof by complete induction the induction hypothesis would have read

   For arbitrarily chosen $n > 0$, for all $m, 0 \leq m \leq n-1$, $\mathbf{P}(m)$

   whereas in the new proof by mathematical induction the induction hypothesis would read

   For arbitrarily chosen $n > 0$, $\mathbf{Q}(n-1)$

   Clearly the two induction hypotheses are logically equivalent. Hence the rest of the proof of the induction step would suffer no other change. The basis step and the induction step would together constitute a proof by *mathematical* induction of the property $\mathbf{Q}$ for all naturals n. Since $\mathbf{Q}(n)$ logically implies $\mathbf{P}(n)$ it follows that the proof of property $\mathbf{P}$ for all naturals has been done by *mathematical* induction.

□

The natural numbers are themselves defined as the smallest set $\mathbb{N}$ such that $0 \in \mathbb{N}$ and whenever $n \in \mathbb{N}$, $n+1$ also belongs to $\mathbb{N}$. Therefore we may state yet another version of PMI from which the other versions previously stated may be derived. The intuition behind this version is that a property $\mathbf{P}$ may also be considered as defining a set $S = \{x \mid xsatisfies\mathbf{P}\}$. Therefore if a property $\mathbf{P}$ is true for all natural numbers the set defined by the property must be the set of natural numbers. This gives us the last version of the principle of mathematical induction.

Principle of Mathematical Induction – Version 0

A set $S = \mathbb{N}$ provided

Basis. *$0 \in S$, and*

Induction step. *For arbitrarily chosen $n > 0$,*
$\underline{n-1 \in S}$ implies $n \in S$.

We end this section with an example of the use of induction to prove that for any $n \in \mathbb{N}$, the set of all n-tuples of natural numbers is only countably infinite.

Example 2.8 *Assume there exists a 1-1 correspondence $f_2 : \mathbb{N}^2 \longrightarrow \mathbb{N}$. Use this fact to prove by induction on n, that there exists a 1-1 correspondence $f_n : \mathbb{N}^n \longrightarrow \mathbb{N}$, for all $n \geq 2$.*

Solution. *In general, to prove that a given function $F : A \longrightarrow B$ is a 1-1 correspondence, we may prove it by contradiction. Then there are 2 cases to consider.*

1. *F is non-injective. Then there exist elements $a, a' \in A$, such that $a \neq a'$ and $F(a) = F(a')$.*
2. *F is non-surjective. Then there exists an element $b \in B$ such that $F(a) \neq b$, for any $a \in A$.*

It is also easy to show that if $F : A \longrightarrow B$ and $G : B \longrightarrow C$ are both bijections then their composition $G \circ F : A \longrightarrow C$ is also a bijection.

We now proceed to prove by induction on n.

Basis. *For $n = 2$ it is given that f_2 is a bijection.*

Induction step. *Assume the induction hypothesis,*

$$\boxed{\text{For some } n \geq 2 \text{ there exists a bijection } f_n : \mathbb{N}^n \longrightarrow \mathbb{N}.}$$

We need to prove that there exists a 1-1 correspondence (bijection) between $\mathbb{N}^{n+1}$ and $\mathbb{N}$. We prove this by constructing a function $f_{n+1} : \mathbb{N}^{n+1} \longrightarrow \mathbb{N}$.

Let $g : \mathbb{N}^{n+1} \longrightarrow (\mathbb{N}^n \times \mathbb{N})$ be a function defined by

$$g(\langle x_1, \ldots, x_n, x_{n+1} \rangle) = \langle \langle x_1, \ldots, x_n \rangle, x_{n+1} \rangle$$

Claim: g is a 1-1 correspondence.

Let $h : \mathbb{N}^{n+1} \longrightarrow (\mathbb{N} \times \mathbb{N})$ be defined by

$$h(\langle x_1, \ldots, x_n, x_{n+1} \rangle) = \langle f_n(\langle x_1, \ldots, x_n \rangle), x_{n+1} \rangle$$

Claim: h is a 1-1 correspondence. It can be proved from the fact that $f_n : \mathbb{N}^n \longrightarrow \mathbb{N}$ is a bijection.

Since f_2 is also a bijection, it follows that the composition of h and f_2, viz. $f_2 \circ h : \mathbb{N}^{n+1} \longrightarrow \mathbb{N}$ is also a bijection. Hence define $f_{n+1} \triangleq f_2 \circ h$, i.e.

$$f_{n+1}(\langle x_1, \ldots, x_n, x_{n+1} \rangle) = f_2(h(\langle x_1, \ldots, x_n, x_{n+1} \rangle))$$

Note.

1. *Many people assume automatically that* $\mathbb{N}^{n+1} = \mathbb{N}^n \times \mathbb{N}$ *or* $\mathbb{N}^{n+1} = \mathbb{N} \times \mathbb{N}^n$*. But while it is true that there exists a bijection between* $\mathbb{N}^{n+1}$ *and* $\mathbb{N}^n \times \mathbb{N}$*, they are not equal as sets. Hence we have defined the function g though it is not really necessary for the proof.*

2. *Very often countability is assumed by people and they try to argue that since the sets are countable there should be a bijection. But it should be clear that estabilishing a bijection is necessary first to prove that the required sets are countable. In fact the aim of this problem is to construct a bijection to prove that the sets* N^n *are all countable.*

2.3 Structural Induction

Definition. Let U be a set called the **Universe**, B a nonempty subset of U called the **basis,** and let K called the **constructor** set be a nonempty set of functions, such that each function f $\in$ K has associated with it a unique natural number n ≥ 0 called its **arity** and denoted by α(f). If a function f has an arity of n ≥ 0, then f : $U^n \ \rightarrow \ U$. A set A is said to be **inductively defined** from B, K, U if it is the smallest set satisfying the following conditions:

1. B $\subseteq$ A and

2. if f $\in$ K is of arity n ≥ 0, $a_1, \ \ldots, \ a_n \in \ A$ *and* $f(a_1, \ \ldots, \ a_n) \ = \ a,$ *then* $a \ \in \ A.$

The set A is also to have been **generated** from the basis B and **the rules of generation** f $\in$ K.

In other words $A \ = \ \cap\{S \ \subseteq \ U \mid S \ satisfied \ conditions$ (1) and (2) $\}$
We may also think of A as a set satisfying the equations
$A \ = \ B \ \cup \ \bigcup\{ \ f(A^n) \mid f \ \in K, \ \alpha(f) \ = \ n \ \geq \ 0\}$
where $f(A^n) \ = \ \{ \ a|a = f(a_1, \ \ldots, \ a_n)\}$, for some $< a_1, \ \ldots, \ a_n > \in A^n\}$

Definition. Let U, B, K be as in the definition. Then a sequence $a_1, \ \ldots, \ a_m$ of elements of U is called a **construction** sequence for a_m if for all i=1, . . . , m either $a_i \ \in$ B or there exists a constructor f $\in$ K, of arity n > 0, and $0 < \ i_i, \ \ldots, \ i_n \ < \ i$ such that $f(a_{i1}, \ \ldots, \ a_{in}) \ = \ a_i$.

A contains all the elements of U which have a construction sequence. In our case the the rules of generation are usually functions rather than relations. The basis along with the constructor functions are said to define a grammar for the of *syntactically correct sentences of the language* A.

Example. Consider the following definition of subclass of arithmetic expressions generated from natural numbers, that is, arithmetic expressions made up only natural numbers and the addition and multiplication operations. The rules may be expressed in English as follows.

(a) every natural number is an arithmetic expression .

(b) If e and e' are arithmetic expressions, then exe' and e+e' are also arithmetic expressions

(c)Only whatever is obtained by repeated applications of (a) and (b) is an arithmetic expressions (and nothing else).

In the above definition of arithmetic expressions the universe consists of all possible strings (i.e. finite sequence) made of natural numbers and the two symbols '+' and 'x'. Let $\sum$ be any collection of symbols we use $\sum^*$ to denote the set of all finite sequences of elements from $\sum$. Hence U = (N $\cup$ $\{+, x\})^*$. The basis is the set N of naturals and the constructor set K consists of two binary functions (i.e. functions of arity 2) viz $f_+(e, e')$ =e+e' and $f_x(e + e') =$ exe'. N $\cup$ K is the language of arithmetic expressions.

A convenient shorthand notation called the *Backus-Naur Form (BNF)* is usually employed to express the rules of generation . For the language of arithmetic expressions defined above the BNF is as follows.

e :: n — e+e — e X e

Where n denotes any natural number and e denotes an arbitrary arithmetic expression (Note that the two occurrence of 'e' in 'e+e' do not denote necessarily the same expressions. Similarly for the two 'e's in exe').

Now consider the arithmetic expressions 2+3*5. This expression could have been obtained by either of the following constructions:

$f_x(f_+(2,\ 3),\ 5)$

$f_+(2,\ f_x(3,\ 5)$

The way we constructs sentences is often important in our understanding of a language. The two orders of construction here also imply that the results of evaluation will differ in the two cases. The first method of construction yields the result 25, whereas the second yields 17. We would say that the given grammar is *ambiguous* because it is possible to generate sentences which do not have unique constructions.

Example 2.9 *The language of minimal logic defined by the following grammar*

$$\mu ::= A \mid (\neg\mu) \mid (\mu \rightarrow \mu)$$

is countable.

Proof: *We first begin classifying the formulas of the language according to their depth. Let M_k be the set of formulas of the language such that each formula has a depth at most k for $k \geq 0$. We assume that $M_0 = \mathbb{A}$ and $M_{k+1} = M_k \cup \{(\neg\mu_k), (\mu_k \rightarrow \nu_k) \mid \mu_k, \nu_k \in M_k\}$. Let*

$\overset{\neg}{M_k}$ be the set of all formulas of depth $k+1$ of the form "$(\neg\mu_k)$" and let $\overset{\rightarrow}{M_k}$ be the set of all formulas of the form "$(\mu_k \rightarrow \nu_k)$", where $\mu_k, nu_k \in M_k$. We then have

$$\begin{aligned} M_{k+1} &= M_k \cup \overset{\neg}{M_k} \cup \overset{\rightarrow}{M_k} \\ &= M_k \cup (\overset{\neg}{M_k} - M_k) \cup (\overset{\rightarrow}{M_k} - M_k) \\ &= M_k \cup N_{k+1} \cup A_{k+1} \end{aligned}$$

Here N_{k+1} represents the set of all formulas of depth exactly *$k+1$, whose root operator is $\neg$. Simlarly A_{k+1} represents the set of all formulas of depth* exactly *$k+1$, whose root operator is $\rightarrow$. Hence the three sets are mutually disjoint.*

$$M_k \cap N_{k+1} = \emptyset, \quad M_k \cap A_{k+1} = \emptyset, \quad N_{k+1} \cap A_{k+1} = \emptyset$$

The entire language may then be defined as the set $\mathcal{M}_0 = \bigcup_{k\geq 0} M_k = \mathbb{A} \cup \bigcup_{k>0} N_k \ \cup \bigcup_{k>0} A_k$

Claim. Each of the sets M_k, N_k and A_k is countably infinite for all $k \geq 0$.
Proof of claim. We prove this claim by induction on k. The basis is $M_0 = \mathbb{A}$ and it is given that it is countably infinite. The induction step proceeds as follows. We have by the induction hypothesis that M_k is countably infinite. Hence there is a $1-1$ correspondence $num_k : M_k \longleftrightarrow \mathbb{N}$. We use num_k to construct the $1-1$ correspondence num_{k+1} as follows: We may use num_k to build a 1-1 correspondence between N_k and $\mathbb{N}$. Similarly there exists a 1-1 correspondence between A_{k+1} and $\mathbb{N} \times \mathbb{N}$ given by the ordered pair of numbers $(num_k(\mu_k), num_k(\nu_k))$ for each $(\mu_k \rightarrow \nu_k) \in A_{k+1}$. But we know that there is a 1-1 correspondence $diag : \mathbb{N} \times \mathbb{N} \longleftrightarrow \mathbb{N}$.

Hence each of the 3 sets M_k, N_{k+1} and A_{k+1} is countably infinite. Their union is clearly countably infinite by the following 1-1 correspondence.

$$num_{k+1}(\mu_{k+1}) = \begin{cases} 3 \times num_k(\mu_{k+1}) & \text{if } \mu_{k+1} \in M_k \\ 3 \times num_k(\mu_k) + 1 & \text{if } \mu_{k+1} \equiv \neg\mu_k \in N_k \\ 3 \times diag(num_k(\mu_k), num_k(\nu_k)) + 2 & \text{if } \mu_{k+1} \equiv \mu_k \rightarrow \nu_k \in A_k \end{cases}$$

Hence N_{k+1} and A_{k+1} are countably infinite from which it follows that M_{k+1} is countably infinite.

Having proved the claim it follows (from the fact that a countable union of countably infinite sets yields a countably infinite set) that M the language of minimal logic is a countably infinite set. □

Since ambiguity can lead to different interpretations ,we would like to avoid using grammars which are ambiguous. One simple way of avoiding ambiguity in arithmetic expressions is to use parentheses. Hence by expanding the set of symbols to include the pair of parentheses '(' and ')', we have

$\mathrm{U} = \mathrm{N} \cup \{ +, \mathrm{x}, (,) \})^*$

and we may redefine the functions f_+ and f_x as follows:

$f_+(e,\ e') \ = \ (e + e')$

$f_x(e,\ e') \ = \ (exe')$

Alternatively the BNF for the new language is

e ::= n | (e+e) | (e x e)

In the new grammar the sentences 2+3x5 is no longer syntactically correct. In other words, though 2+3x5 does belong to the universe U, it does not belong

to the set A by the inductive definition. However, in its place we have the two new sentences ((2+3)x5) and (2+(3x5)). Each of these sentence has a unique construction viz .

$((2+3)x5) \ = \ f_x(f_+(2,\ 3),\ 5) \ (2+(3x5)) = f_+(2,\ f_x(3,\ 5))$

We are now ready to formally define when a language that is inductively defined is unambiguous.

Definition. Let T ⊆ U be inductively defined by a basis B and constructor set K. A is said to be free (or unambiguous) if for every element a ∈ A, either A ∈ B or there exists a unique constructor f ∈ K and a unique $\alpha(f)$-tuple $< a_1,\ \ldots, a_{\alpha(f)} > \ \in \ A^{\alpha(f)}$ such that $f(a_1,\ \ldots, a_{\alpha(f)})$ = a.

Structural Induction.

We present below a generalization of the principal of mathematical induction to arbitrary inductively defined sets. It provides us a way of reasoning about the properties of structures that are inductively defined.

Theorem. *The Principle of structural induction (PSI).* Let A ⊆ U be inductively defined by the basis B and the constructor set K. Then a property P holds for all elements of A provided

(i) **Basis. P** is true for all basis elements.

(ii) **Induction step.** For each f ∈ K, if **P** holds for elements $a_1,\ \ldots, a_{\alpha(f)} \ \in \ A$ and $f(a_1,\ \ldots, a_{\alpha(f)}$ = a, then P true for a.

Proof. Let C be the set of all elements of A that satisfy the property **P**,i.e. C = { a ∈ A | P holds for a}. It is clear that C ⊆ A. It is therefore only necessary to show a ⊆ C. Clearly from the inductive basis B ⊆ C. Also from (ii) it is clear that for any $a_1,\ \ldots, a_{\alpha(f)} \ \in \ C$ and $f(a_1,\ \ldots, a_{\alpha(f)}$ = a that a ∈ C. Hence A ⊆ C and therefore A = C.

Example. Consider the following grammar of arithmetic expressions

e ::= n | (e+e) | (exe) | (e-e)

Let e be any expression and P be property

For every prefix[1] *e' of e the number of left parentheses in e' is greater than or equal to the number of right parentheses.*

This property holds for all expressions e in the language . P may be proved by PSI as follows:

[1] Given a string w of symbols, a string u is called a prefix of w if there is a string v such that $w = u.\ v$, where '.' denotes the just a position(or catenation) operation on strings. Clearly the empty string ϵ is a prefix of every string and every string is a prefix of itself. u is called a proper prefix of w if v is a nonempty string.

Basis. It holds for all n $\in$ N because no natural number has any parentheses.
Induction step. Let e be any expression not in the basis. Then e has only one of the following forms viz. f $\odot$ g, where $\odot$ $\in$ {+,x,-} and f, g are themselves expressions in the language. Then we may assume the following.
Induction hypothesis.
(a) For every prefix f' of f , the number of left parentheses in f' is greater than or equal to the number of right parentheses.
(b) For every prefix g' of g the number of right parentheses in g' is greater than or equal to the number of right parentheses.

We now have to show that P holds for e = f $\odot$ g. that is, we need to show that for every prefix e' of e, the number of left parentheses in e' is at least as many as the number of right parentheses in e'. For this it is necessary to consider the following cases, which are essentially all the possible prefixes e' of e
Case (i) e'= ε
Case (ii) e'=(
Case (iii) e'=(f',where f' is a prefix of f
Case (iv) e'=(f $\odot$
Case (v) (f $\odot$ g', where g' is a prefix of g
Case (vi) e'=(f $\odot$ g)
For each string e' (which may be a prefix of a string of the language) Let L(e') and R(e') denote respectively the numbers of left and right parentheses in e'. By the induction hypotheses we may assume that property P holds for both f and g, i.e. for every prefix f' of f and g' of g, we have the following inequalities.
(a) L(f') $\geq$ R(f') (b) L(g') $\geq$ R(g')
We prove *case(v)* and leave the rest of cases to be proved by the reader. We have e'=(f $\odot$ g' where g' is a prefix of g. This yields the following

$$\begin{aligned} R(e') &= R(f) + R(g') \\ L(e') &= 1 + L(f) + L(g') \\ &> L(f) + L(g') \\ &\geq R(f) + R(g') \\ &= R(e') \end{aligned}$$

hence L(e') $\geq$ R(e')

We are now ready to show that even though structural induction seems to be more general than mathematical induction,the are in fact equivalent. We leave it as an exercise for the reader to prove that every proof by the principal of structural induction. To do this we need to define the notion of depth of a member A, where A is an inductively define set.

However, we do realize that if A is not free then an element of A may not have a unique depth. To this end we define the following.

Definition. Let A $\subseteq$ U be a set inductively defined by a basis B and a constructor set K. For each construction c(a) of a define the natural number $\triangle$(c(a)), called **the depth of construction** of a as follows.

$\triangle$(c(a)) = 0 for every element of the basis B.
if c(a) = f($c_1(a_1)$, $\ldots, c_n(a_n)$) for some n-ary (n $\geq$ 0) constructor f and elements $a_1, \ldots, a_n$ $\in$ A, then

$$\triangle(c(a)) = 1 + \max(\triangle(c_n(a_1)), \ldots, \triangle(c_n(a_n))).$$

Since an element of the set may have several different possible constructions, $\triangle$(a) may have different values, though for a particular construction it would be unique.

Definition. Let A $\subseteq$ U be a set inductively defined by a basis B and a constructor set K. For every element a $\in$ A, the **depth** of a is defined as a function δ : A $\rightarrow$ N, such that

δ(a) = min $\triangle$(c(a)) | c is a construction of a

Theorem. Every proof using PSI may be replaced by a proof using PMI.
proof. Let A be inductively defined by B, K, U and let P be a property of elements of A that has been proved by PSI.
Let the property Q(n) for each natural number n be defined as

The property P holds for all elements of a depth n

Then it is clear that the proof by PSI for each element of the basis is a proof of the basis step for Q(0) and the induction hypothesis is the assumption
Q(m) holds for all 0 $\leq$ m $<$ n.
The induction step is simply that if the induction hypothesis holds then Q holds for n. The proof by PSI for each constructor in the induction step is a case for Q(n).

Inductive definitions of functions
Definition. Let a $\subseteq$ U be inductively defined by B, K, and let V be any arbitrary set. Let
h: B $\rightarrow$ V
be a function and with every n-ary constructor f $\in$ K, let
H(f): V^n $\rightarrow$ V
be an n-ary function. Then a relation g $\subseteq$ A x V is said to be **inductively defined** if g(b) = h(b) for every b $\in$ B and g(a) = H(f)(b(a_1), $\ldots, g(a_n)$) whenever $f(a_1, \ldots a_n)$ = a.

Proposition. If a $\subseteq$ U in the above definition is free (unambiguous) then the relation g given above is well defined (i.e. g is a function g:A $\rightarrow$ V).
proof. Assume A is free and let g be as in the above definition, then we have to show that for all a $\in$ A, g(a) is unique. We prove this by PSI on A.

Since for all b ∈ B, there is no other way of constructing b, it follows that g(b) = h(b) is the only image of b ∈ B. Suppose $f(a_1, \ldots, a_n)$ = a ,for some n-ary constructor f ∈ K. By the induction hypothesis $g(a_1)$, . . . , $g(a_n)$ all have unique values. Since A is unambiguous there is no other way of constructing a and there is a unique definition of g(a) given by g(a) = H(r) $(g(a_1), \ \ldots, g(a_n))$.

Note. This proposition on the inductive method of defining functions holds good even if f ∈ K is not a function. For if $<< \ a_1, \ \ldots, a_n >, \ a >$ and $<< a_1, \ \ldots, a_n >, \ a' >$ both belong to a relation f ∈ K, Then g(a) = H(r)$(g(a_1), \ \ldots, g(a_n))$ = g(a'), which makes g a well defined function.

Also it is clear that if A is free then every element of a has a unique construction and therefore the function △ and δ coincide for each element.

Example. Consider the language L_0 of propositional logic, where the basis B is the set of atoms, (use capital letters like P, Q, R), and the language is defined by the BNF

p ::= P | (¬ p) | (p ∨ p) | (p ∧ p) | (p → p) | (p ↔ p)

Further let B = <{0,1}, . . .,+,-> be the algebraic system whose operations are defined as follows.

0 . 0 = 0 . 1 = 1 . 0 = 0 and 1 . 1 = 1

1 + 1 = 0 + 1 = 1 + 0 = 1 and 0 + 0 = 0

$\bar{0}$ = 1 and $\bar{1}$ = 0

Let τ : B → {0,1} be a truth value assignment for the propositional names in B. Then the interpretation of propositional formulas in the truth assignment τ, is the function

$\mathcal{T} : L_0 \rightarrow \{0, 1\}$
obtained by inductively extended τ as follows.
$\mathcal{T}[\mathrm{P}] = \tau[\mathrm{P}]$
$\mathcal{T}[(\neg\ \mathrm{p})] = \overline{\mathcal{T}[P]}$
$\mathcal{T}[(\mathrm{p} \wedge \mathrm{q})] = \mathcal{T}[\mathrm{p}].\mathcal{T}[\mathrm{q}]$
$\mathcal{T}[(\mathrm{p} \vee \mathrm{q})] = \mathcal{T}[\mathrm{p}] + \mathcal{T}[\mathrm{q}]$
$\mathcal{T}[\mathrm{p} \rightarrow \mathrm{q})] = \overline{\mathcal{T}[\mathrm{p}]} + \mathcal{T}[\mathrm{q}]$
$\mathcal{T}[(\mathrm{p} \leftrightarrow \mathrm{q})] = \mathcal{T}[\mathrm{p}].\mathcal{T}[\mathrm{q}] + \overline{\mathcal{T}[\mathrm{p}].\mathcal{T}[\mathrm{q}]}$
We have assumed throughout that '.' has precedence over '+' and '-' has precedence over '.'. It can be shown that $\mathcal{T}$ is a well defined function since the set PF_B is free.

2.4 Exercises

1. Prove that version 1, 2 and 3 of PMI are mutually equivalent.

2. Find the fallacy in the proof of the following purported theorem.
   Theorem: If X = Y then 2 = 1.

(a) X=Y	; Given
(b) X^2=XY	; Multiply both sides by X
(c) $X^2 - Y^2 = XY - Y^2$	; Subtract Y^2 from both sides
(d) (X+Y)(X-Y)=Y(X-Y)	; Factor
(e) X+Y=Y	; Cancel out (X-Y) term
(f) 2Y=Y	; Substitute X for Y, by equation 1
(g) 2=1	; Divide both sides by Y

3. Find the fallacy in the following proof by PMI. Rectify it and again prove using PMI.
Theorem:
$$\frac{1}{1*2} + \frac{1}{2*3} + \ldots + \frac{1}{(n-1)n} = \frac{3}{2} * \frac{1}{n}$$
Proof: For n=1 the LHS is 1/2 and so is RHS. assume the theorem is true for some $n > 1$. We then prove the induction step.
$$\begin{aligned} LHS &= \frac{1}{1*2} + \frac{1}{2*3} + \frac{1}{(n-1)n} + \ldots + \frac{1}{n(n+1)} \\ &= \frac{3}{2} - \frac{1}{n} + \frac{1}{n(n+1)} \\ &= \frac{3}{2} - \frac{1}{n} + \frac{1}{n} - \frac{1}{(n+1)} \\ &= \frac{3}{2} - \frac{1}{n(n+1)} \end{aligned}$$
which is required to be proved.

4. Let A be any set with a reflexive and transitive binary relation $\preceq$ defined on it. That is to say, $\preceq \subseteq$ A x A satisfies the following conditions.

   (a) For every $a \in A$, $a \preceq a$.

   (b) For all a, b, c in A, $a \preceq b$ and $b \preceq c$ implies $a \preceq c$.

   Then show by induction that $\preceq \circ R = \preceq^n \circ R$ for all n $\geq$ 1.

5. Show using PSI that $\mathcal{T}$ is well defined.

6. The language of numerals in the arabic notation may be defined by the following simple grammar
d ::= 0|1|2|3|4|5|6|7|8|9
n ::= $d|nd$
Define the value of a string in this language, so that it conforms to the normally accepted meaning of a numeral

7. For the language of propositions we may also call the function τ a state, and look upon the function $\mathcal{T}$ as defining the truth value of a proposition in a given state τ. Now redefine the meaning of a proposition as the set of states in which the proposition has a truth value of 1. if Σ denotes the set of all possible states i.e. $\Sigma = \{\tau | \tau : B \rightarrow \{0,1\}\}$ then

(a) What is the domain and the range of the function φ?

(b) Define the function φ by structural induction.

(c) Prove using the principle of structural induction that for any proposition p, and a state τ, $\mathcal{T}$(p) = 1 if and only if *tau* belongs to the φ-meaning of p.

8. Let A be a language ,inductively defined by B, K, U. Define the set of *syntax tree*, T_A of the elements of A as follows:

   (a) For each b $\in$ B, there is a single node labelled b,

   (b) For each n-ary operator $\odot$ and $a_1, \ldots, a_n, a \in A$, if $\odot(a_1, \ldots, a_n) = a$, the syntax tree t of a is a tree with root labelled by $\odot$ and $t_1, \ldots, t_n$ as the immediate subtree of t, where $t_1, \ldots, t_n$ are the syntax trees corresponding to $a_1, \ldots, a_n$ respectively.

   (a) Prove that every element of A has a unique syntax tree if A is free.

   (b) Give an example to show that every syntax tree need not define a unique element of A.

9. Let L_0 be the language of propositional logic as defined in the last example. Then intuitively speaking, a propositional formula p is a **subformula** of a propositional formula q if the syntax tree of p is a subtree of the syntax tree of q.

   (a) Define the notion of subformula inductively.

   (b) Let of every formula q, SF(q) denote the set of all subformulas of q. Define SF(q) inductively.

   (c) Let p $\preceq$ q if and only if p is a subformula of q. Prove that $\preceq$ is a partial order on L_0.

Chapter 3

Propositional Logic

"Contrariwise", continued Tweedledee, "if it was so, it might be, and if it were so, it would be; but as it isn't, it ain't. That's logic!"

Lewis Carroll, "Through the Looking Glass"

3.1 Syntax of Propositional Logic

In this section we will use structural induction to define the syntax and semantics of *propositional* or *sentential* logic.

Definition 3.1 *Let*

- $\mathbb{A}$ *be a countably infinite collection of propositional* atoms, [1]
- $\mathbb{K} = \{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\}$ *be the set of* operators *with each operator being assigned a unique* arity[2] *defined by a function* $\alpha : \mathbb{K} \rightarrow \mathbb{N}$*, such that* $\alpha(\neg) = 1$, $\alpha(\odot) = 2$*, for* $\odot \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$ *and*
- $\mathbb{A}$ *and* $\mathbb{K}$ *be disjoint sets.*

The language of propositional logic $\mathcal{P}_0$ *is the smallest set generated from* $\mathbb{A}$ *and* $\mathbb{K}$*.*

[1] You may think of these atoms as playing a role similar to that of variables in any programming language.

[2] The arity of an operator simply signifies the number of operands that the operator requires. For example, addition on numbers is a binary operation and hence its arity is 2. Operations on numbers such as $\sum$ are unary (since they take a single operand such as a set or a sequence of numbers). Mathematically speaking, no operator can have an undefined arity or infinite arity.

Alternatively we may define the language of propositional logic as the set of sentences generated by the following grammar. This has the advantage that it allows us to also specify the other conventions — such as precedence of operators, bracketing conventions etc. — while writing and expressing propositions.

Definition 3.2 *Let*

- $\mathbb{A}$ *be a countably infinite collection of propositional* atoms*,*
- $\mathbb{K} \cup \{(,)\}$ *be disjoint from* $\mathbb{A}$*,*
- $\mathbb{A} \cup \mathbb{K} \cup \{(,)\}$ *be the set of terminal symbols,*
- ϕ *be the single non-terminal symbol (also the start symbol of the grammar) and*

$$\phi ::= A \mid (\neg\phi) \mid (\phi \wedge \phi) \mid (\phi \vee \phi) \mid (\phi \rightarrow \phi) \mid (\phi \leftrightarrow \phi)$$

be the set of production rules where $A \in \mathbb{A}$*.*

Then the language of propositional logic is the set of sentences generated by the above grammar. The sentences of this language are also called well- formed formulas *or* wffs *of propositional logic.*

Example 3.1 *The following are wffs of propostional logic, assuming that* A*,* B*,* C*,* D *are atoms taken from the set* $\mathbb{A}$*.*

- $(\neg A)$
- $(A \wedge B)$
- $(\neg((A \rightarrow B) \rightarrow (C \leftrightarrow D)))$
- $((A \vee D) \wedge ((\neg C) \leftrightarrow D))$

Example 3.2 *The following are* not *wffs of propositional logic (*Why not?*)*

- $((A \vee \leftrightarrow) \wedge \neg C$
- $((A \vee D) \wedge (\neg C \leftrightarrow D))$
- $\neg((A \rightarrow B) \rightarrow (C \leftrightarrow D))$
- $((A \vee D) \wedge ((\neg C) \leftrightarrow D))$
- $(((A \vee D) \wedge ((\neg C) \leftrightarrow D)))$

The grammar above tells us that binary operators are all to be used in *infix*[3] form and the negation operator ($\neg$) is to be used in *prefix form* [4].

We have used a fully bracketed syntax for propositions. This can make it quite cumbersome to write propostions with so many parentheses. We shall however, we be much more relaxed while writing out propositions by using the following precedence conventions:

Convention 3.1

$$\leftrightarrow \; < \; \rightarrow \; < \; \vee \; < \; \wedge \; < \; \neg$$

is the increasing order of precedence of the operators. This implies that $\neg$ *binds the tightest and* $\leftrightarrow$ *binds the loosest.*

Hence a propositional formula maybe written without parentheses provided it is clear what the operands of each operator are, by applying the convention. And of course, we could always override convention by using parentheses to enclose the appropriate entitites.

Example 3.3 *Consider the following.*

1. $(A \vee B) \wedge \neg C$*. This clearly stands for* $((A \vee B) \wedge (\neg C))$*. The parentheses around* $\neg$ *have been removed and so have the outermost pair of parentheses. The parentheses around* $A \vee B$ *are necessary if we want the "*$\vee$*" between* A *and* B *to bind more tightly than the "*$\wedge$*" between* B *and* $\neg C$*.*

2. *On the other hand, the formula* $A \vee B \wedge \neg C$*, without any parentheses, would be interpreted as being the same as* $(A \vee (B \wedge (\neg C)))$*.*

3.2 The model of truth

For any formal language it is important to be able to give a meaning to the symbols used in the language. In general any formal language consists of variables symbols, constant symbols, operators and punctuation symbols. The punctuation symbols (such as "(" and ")") do not carry any meaning in themselves; their role is restricted to ensuring readability and to aid in parsing a sentence of the language. However, the other symbols in the language do carry with them a meaning.

The notion of meaning itself may vary widely depending upon the kind of formal language that is of interest to us. For example if the formal language is a programming language, then meaning refers usually to the effect the various constructs have on the values being processed by a program. In a logic, the notion of meaning is usually restricted to whether a sentence is

[3] That is, the operator appears between its operands

[4] the operator always precedes its operand

true or false. This primarily is the role of logic in mathemtatics. Ultimately, in any logical formalization of a mathematical theory we are interested in how to reason about the "objects" of the theory. We shall take the same view about the logical languages that we study in this course. In the case of formal language models of natural languages (such as English) the notion of meaning is much deeper and much harder to capture, since most sentences in such languages usually refer to objects in the real-world. By a logic therefore, we refer to a formal language and its *semantics* which along with notions of truth and falsity, define the meaning of each sentence of the formal language.

While restricting our attention to the notion of truth (and falsity), we shall take a rather simplistic view. We simply assume that any sentence in the language could either be true or false[5]. This simplistic notion turns out to be quite powerful in modelling reasoning methods which underlie a large part of mathematics. There are however systems of "multiple-valued logics" and "fuzzy logics" where absolute truth and absolute falsity are merely the two extreme points of a spectrum of truth and falsity[6].

In defining the semantics of propositional logic, we use the boolean algebra of truth values $\{0, 1\}$ where 0 represents "false" and 1 represents truth. We also take the standard operations of boolean algebra and augment it with two more operations.

Definition 3.3 *Let* $\mathcal{B} = \langle \{0, 1\}, \{\bar{\ }, ., +\}, \{\leq, =\} \rangle$ *be the* 2*-element boolean algebra where the operations take their usual meanings. In other words, for* $a, b \in \{0, 1\} \subseteq \mathbb{N}$*,*

- $\bar{a} = 1 - a$ *is the unary boolean* <u>inverse</u> *operation,*
- $a + b = max(a, b)$ *is the binary* <u>summation</u> *operation which yields the maximum of two boolean values regarded as natural numbers,*
- $a.b = min(a, b)$ *is the binary* <u>product</u> *operation which yields the minimum of two boolean values regarded as natural numbers,*
- $a \leq b$*, and* $a = b$ *denote the usual binary relations "less-than-or-equal-to" and "equals" on natural numbers restricted to the set* $\{0, 1\}$*.*

While $\leq$ and $=$ are binary relations, it is possible to define corresponding binary operations as shown below.

Definition 3.4 *For* $a, b \in \{0, 1\}$ *define*

$$a \leq\cdot\, b = \begin{cases} 1 & \textit{if } a \leq b \\ 0 & \textit{otherwise} \end{cases} \qquad a \doteq b = \begin{cases} 1 & \textit{if } a = b \\ 0 & \textit{otherwise} \end{cases}$$

[5] For instance, we assume there are no "shades of grey"

[6] There are "shades of grey" in such logics. In a multiple valued logic the spectrum comprising truth and falsity consists of a finite collection of values (e.g. the set $\{0, 1, 2, \ldots, n-1\}$), usually linearly ordered according to the amount of truth or falsity that a value denotes. A fuzzy logic on the other hand uses a continuum, e.g. the closed real interval $[0, 1]$

Armed with these operations we are now ready to define the semantics of propositional logic.

3.3 Semantics of Propositional Logic

Our notion of meaning is restricted to the two truth values 0 and 1 of the boolean algebra $\mathcal{B}$ denoting falsity and truth respectively.

Definition 3.5 *Let $\tau : \mathbb{A} \rightarrow \{0, 1\}$ called a* truth assignment*, be a function which associates a truth value with each atom in $\mathbb{A}$. For each atom A under a truth assignment τ, $\tau(A)$ yields a value 0 if A is false and 1 if it is true.*

Then meaning *(of a proposition ϕ) under the truth assignment τ is a function*

$$\mathcal{T}\llbracket\bullet\rrbracket_\tau : \mathcal{P} \rightarrow \{0, 1\}$$

defined by induction on the structure of the proposition as follows.

$$\begin{array}{lcllcl}
\mathcal{T}\llbracket A\rrbracket_\tau & \triangleq & \tau(A) & \mathcal{T}\llbracket \neg\phi\rrbracket_\tau & \triangleq & \overline{\mathcal{T}\llbracket\phi\rrbracket_\tau} \\
\mathcal{T}\llbracket \phi \wedge \psi\rrbracket_\tau & \triangleq & \mathcal{T}\llbracket\phi\rrbracket_\tau . \mathcal{T}\llbracket\psi\rrbracket_\tau & \mathcal{T}\llbracket \phi \vee \psi\rrbracket_\tau & \triangleq & \mathcal{T}\llbracket\phi\rrbracket_\tau + \mathcal{T}\llbracket\psi\rrbracket_\tau \\
\mathcal{T}\llbracket \phi \rightarrow \psi\rrbracket_\tau & \triangleq & \mathcal{T}\llbracket\phi\rrbracket_\tau \leq\cdot\ \mathcal{T}\llbracket\psi\rrbracket_\tau & \mathcal{T}\llbracket \phi \leftrightarrow \psi\rrbracket_\tau & \triangleq & (\mathcal{T}\llbracket\phi\rrbracket_\tau \doteq \mathcal{T}\llbracket\psi\rrbracket_\tau)
\end{array}$$

We use the symbol $\triangleq$ to denote that these are *definitions* and not mere equalities[7].

3.4 Satisfiability, Validity and Contingency

Definition 3.6 *A truth assignment τ* satisfies *a propositional formula ϕ if $\mathcal{T}\llbracket A\rrbracket_\tau = 1$. We denote this fact by $\tau \models \phi$. Simlarly $\tau \not\models \phi$ denotes the fact the τ does not satisfy ϕ. A formula is said to be* satisfiable *only if there exists a truth which satisfies it.*

Example 3.4 *Consider three atoms A, B, C, and the formula $A \wedge B \rightarrow C$.*

1. *Let τ_0 be a truth assignment in which all the three atoms are true. Clearly $\tau_0 \models A \wedge B \rightarrow C$,*
2. *Let τ_1 be a truth assignment in which all the three atoms are false. Again $\tau_1 \models A \wedge B \rightarrow C$,*

[7] A definition is also an equality and hence the left hand sides and right handsides of a definition can replace each other always. However, they are equalities by definition and are not idenitites that are derived by other means. For example an identity such as $(a + b)(a - b) = a^2 - b^2$ is an equality but is not a definition

3. *Let τ_2 be a truth assignment in which A and C are true. Clearly $\tau_2 \models A \wedge B \rightarrow C$,*

4. *Let τ_3 be a truth assignment in which one of A and B is true and C is false. Clearly $\tau_3 \not\models A \wedge B \rightarrow C$,*

5. *The formula $A \wedge B \rightarrow C$ is clearly satisfiable.*

Definition 3.7 *A formula is said to be* unsatisfiable *if there exists no truth assignment that can satisfy it. An unsatisfiable proposition is also called a* contradiction.

Example 3.5 *The following formulas are unsatisfiable (assume A is an atom)*

1. $A \wedge \neg A$

2. $\neg(A \vee \neg A)$

Definition 3.8 *A proposition $\phi \in \mathcal{P}_0$ is said to be* **logically valid** *(denoted by $\models p$) , iff for every truth assignment τ, $\tau \;\models\; p$.*

Definition 3.9 *A proposition $\phi \in \mathcal{P}_0$ is said to be* **unsatisfiable** *(or a* **contradiction***), iff for every truth assignment τ, $\mathcal{T}[\![\phi]\!]_\tau = 0$.*

Definition 3.10 *A proposition is said to be* **contingent** *if it is neither a valid nor unsatisfiable.*

A contingent statement is true for some (atleast one) truth assignment and false for some other (atleast one) truth assignment.

Theorem 3.2 *Let $\Gamma = \{\phi_i | 1 \leq i \leq n\}$ be a finite set of propositions, and let ψ be any proposition*[8]*. Then $\Gamma \models \psi$ <u>if and only if</u> $((\ldots((\phi_1 \wedge \phi_2) \wedge \phi_3) \wedge \ldots \wedge \phi_n) \rightarrow \psi)$ is a tautology.*

Proof:

($\Rightarrow$) Suppose $\Gamma \models \psi$. Logical consequence guarantees that for any truth assignment τ, if

$$\mathcal{T}[\![\phi_1]\!]_\tau = \ldots = \mathcal{T}[\![\phi_n]\!]_\tau = 1$$

then $\mathcal{T}[\![\psi]\!]_\tau) = 1$

[8] ψ could also be one of the propositions in Γ

Now assume $((\ldots((\phi_1 \wedge \phi_2) \wedge \phi_3) \wedge \ldots \wedge \phi_n) \rightarrow \psi)$ is not a tautology. Then there exists a truth assignment τ such that

$$\begin{array}{rcccccc}
& \mathcal{T}[\![((\ldots((\phi_1 \wedge \phi_2) \wedge \phi_3) \wedge \ldots \wedge \phi_n) \rightarrow \psi)]\!]_\tau & = & 0 & & & \\
\textit{i.e.} & (\mathcal{T}[\![(\ldots((\phi_1 \wedge \phi_2) \wedge \phi_3) \wedge \ldots \wedge \phi_n)]\!]_\tau \leq \mathcal{T}[\![\psi]\!]_\tau) & = & 0 & & & \\
\textit{i.e.} & (\mathcal{T}[\![(\ldots((\phi_1 \wedge \phi_2) \wedge \phi_3) \wedge \ldots \wedge \phi_n)]\!]_\tau & = & 1 & \textit{and} \quad \mathcal{T}[\![\psi]\!]_\tau) & = & 0 \\
\textit{i.e.} & \mathcal{T}[\![\phi_1]\!]_\tau = \ldots = \mathcal{T}[\![\phi_n]\!]_\tau & = & 1 & \textit{and} \quad \mathcal{T}[\![\psi]\!]_\tau) & = & 0
\end{array}$$

which contradicts the notion of logical consequence.

$(\Leftarrow)$ Assume $((\ldots((\phi_1 \wedge \phi_2) \wedge \phi_3) \wedge \ldots \wedge \phi_n) \rightarrow \psi)$ is a tautology, and suppose $\Gamma \not\models \psi$. Then there exists a truth assignment τ such that

$$\mathcal{T}[\![\phi_1]\!]_\tau = \ldots = \mathcal{T}[\![\phi_n]\!]_\tau \quad = \quad 1 \quad \textit{and} \quad \mathcal{T}[\![\psi]\!]_\tau) \quad = \quad 0$$

By working backwards over the steps of the previous proof, we obtain

$$\mathcal{T}[\![((\ldots((\phi_1 \wedge \phi_2) \wedge \phi_3) \wedge \ldots \wedge \phi_n) \rightarrow \psi)]\!]_\tau = 0$$

from which it follows that $((\ldots((\phi_1 \wedge \phi_2) \wedge \phi_3) \wedge \ldots \wedge \phi_n) \rightarrow \psi)$ is not a tautology, contradicting our assumption.

□

Chapter 4

An Axiomatic Theory for Propositional Logic

4.1 Introduction

The truth table technique that we have described to check out whether an argument is valid is a rather cumbersome process, especially when the number of atomic proposition is large. Even when an argument consists of just 6 atomic propositions we require a truth table with 64 rows! That means that with the truth table method we can only verify but small arguments. Further the number of columns that have to be constructed depends upon the complexity of the propositions involved. For each operator in the propositions a new column is required. Therefore the size of the truth table increases exponentially with the number of atomic propositions. Other methods such as semantic tableaus and resolution do provide more efficient and automated ways of proving the validity of arguments, finding satisfying valuations etc.

Even more importantly, these model-checking methods are useful only for propositional arguments. As we will see in the chapters on First Order Logic, it is impossible to construct truth tables in all but the most trivial mathematical theories. This is because the truth tables or semantic tableau could be infinitary. When faced against such infinitary structures, model-checking methods are neither adequate nor useful.

However, while model-checking methods possess the advantage that they are fast, simple and can be automated easily on a modern day personal computer, they lack one important property – viz they are nowhere near modelling the process of reasoning as understood in mathematics. They are only useful for validating or falsifying arguments. After all, the main purpose of logic is to formalize the process of reasoning, which allows one to use certain rules of logical deduction to obtain conclusions that may be considered logically valid given certain assumptions. Also it should disallow the use of invalid argument forms. As pointed out in the previous paragraph, when we have to reason about structures that are infinitary in nature there are no methods available except the method of deduction.

It would therefore be nice if we we could perform deductions using a small set of sound argument forms. This set of the argument forms could be proved valid using the truth table technique. However, it should not be necessary to resort to truth tables or to the semantics of the argument in order to infer further facts form a given set of assumptions. It should be possible to obtain conclusions by simply "looking" at the *forms* of the assumptions by a set of rules which allow fresh conclusions to be drawn from the "patterns" of propositions. These rules, being sound, allow us to draw valid conclusions, without using the "meaning" of the assumptions in any way. Very often the notion of "meaning" in such systems is itself infinitary in nature.

Such a system of valid argument forms is called a **deductive** system (or **proof**) system. By its very nature it prohibits the construction and the use of meaning in drawing correct conclusions from a collection of assumptions. It allows only for small set of *pattern matching* and *pattern substitution rules.*

A system based entirely on pattern matching and substitution has certain advantages. Firstly, it is very general since it does not have to concern itself with underlying meanings. Secondly, it is likely to be feasible to implement it algorithmically on machines, since the process of matching and substitution is well understood by programmers and efficient algorithms do exist. However as we shall see, for any given problem there may be many different ways of drawing the same valid conclusion. In other words, there may be many different proofs. In such a case any automatic system that implements the proof system, is best looked upon as a device to "check" the correctness of a given proof rather than as a system to "create" a proof of a theorem. In certain restricted cases, (especially where proof strategies are also programmed into the system) we might be able to "create" or "generate" automatic proofs. Such systems are called proof-assistants.

If a proof system is so constructed then it is clear that it should allow only valid arguments to be proven. Secondly, if it allows for the construction of complex valid arguments from simpler ones, then clearly it should preserve truth. Needless to say, if it is truth preserving then it should also preserve validity or (to coin a new term) "tautologous-ness". Therefore using combinations of simple valid argument forms can give us more complex valid argument forms, and all of them would be truth preserving.

In addition of course, such a system must be **consistent**, that is, it should not enable us to prove contradictions unless the set of assumptions itself is inconsistent. Further it is desirable that a deductive system be sufficiently powerful, so that all possible truth preserving arguments can be proven. In other words we are looking for a deductively **complete system**.

In this chapter we define and prove some important properties of proof systems for propositional logic. To keep the exposition simple, we use only a small subset of the original language of the propositional logic that was previously defined. In other words we use only two connectives initially. As will be evident later the other connectives may be defined in terms of the two that we shall be using. We introduce the concepts of formal theories and proof systems. We then define a formal theory of propositional logic and prove its consistency, completeness and decidability. We also demonstrate the correctness of some important proof methods that are used throughout mathematics and indeed in all forms of reasoning.

4.2 Formal theories

We define the notion of a formal theory. But briefly, a theory consists of some set of objects (they could be mathematical or merely terms of a some formally defined language), a collection of axioms and some rules of inference which allow us to prove properties about the objects of the interest. A formal theory is one which satisfies certain conditions relating to whether it is possible to verify certain properties algorithmically. We describe the idea of formal theory in the following definition.

Definition 4.1

A formal theory $\mathcal{T}$ is said to be defined if it consists of the following.

1. A (countable) *language* $\mathbb{L}$ inductively defined from a countable collection of symbols, such that the **well-formed formulas** or **wffs** of the language may be clearly separated from other strings generated from the alphabet. More accurately, there exists an algorithm which can determine for any string of symbols from the alphabet whether the string belongs to the language. We say membership in the language is **decidable** if such an algorithm exists. These symbols may be any collection of objects. In the case of propositional logic, we define the set of symbols to consist of all the atoms, the operators and the parenthesis symbols, "(" and ")". As usual we use the symbol $\mathbb{L}_0$ to denote the language of propositions.

2. A set $\mathcal{A} \subseteq \mathbb{L}$ of **axioms**. Usually a subset of the wffs is the set aside and called the axioms of the theory. This set could be empty, nonempty but finite or even infinite. In fact, in one of the proof systems we study, it is going to be empty.

3. A finite set of **inference rules**. Each inference rule $\mathcal{R}$ of arity j for some $j \geq 0$ is a relation such that $\mathcal{R} \subseteq \mathbb{L}^j \times \mathbb{L}$ and there exists an algorithm to decide membership in the relation. In other words, for each rule of inference $\mathcal{R}$, there exists a unique natural number j, such that $\mathcal{R} \subseteq \mathbb{L}^j \times \mathbb{L}$ and there exists an algorithm such that for any pair $\langle\langle p_1, \ldots, p_j\rangle, q\rangle$ the algorithm will determine whether the given pair belongs to $\mathcal{R}$. If $\langle\langle p_1, \ldots, p_j\rangle, q\rangle \in \mathcal{R}$ then we say that q is a **direct consequence** of $p_1, \ldots, p_j$ by **virtue of** $\mathcal{R}$. $p_1, \ldots, p_j$ are called the **premises** (or the **hypotheses**) and q is called the **conclusion**. We also say that the conclusion **directly depends upon** the premises. If $j = 0$ then the inference rule is called an **axiom schema**.

We anticipate the notions of axiom schemas and rules of inference in propositional logic, by using some of them as examples.

Example 4.1 *Consider the following axiom schema.*

$$\mathsf{X} \to (\neg\mathsf{X} \to \mathsf{Y}) \tag{4.1}$$

It represents the set of all propositions that satisfy the structural constraint defined by the schema. That is, this single schema represents the set of propositions

$$\{p \to (\neg p \to q) \mid p, q \in \mathbb{L}_0\} \tag{4.2}$$

The formulas in this set have the "shape" defined by the pattern in the schema 4.1. That is for any propositions p and q, we say that $p \to (\neg p \to q)$ satisfies the shape defined by $\mathsf{X} \to (\neg\mathsf{X} \to \mathsf{Y})$. Let $p \equiv A \to \neg B$ and $q \equiv B \to (\neg A \to B)$, where A and B are atoms. Then $(A \to \neg B) \to (\neg(A \to \neg B) \to (B \to (\neg A \to B)))$ belongs to the set in (4.2) defined above by the schema in (4.1).

Example 4.2 *Consider the following simple inference rule.*

$$\frac{X \to Y, Y \to Z}{X \to Z} \tag{4.3}$$

This 2-ary inference rule which essentially captures the transitivity of the conditional, represents the set

$$\{\langle\langle p \to q, q \to r\rangle, p \to r\rangle \mid p, q, r \in \mathbb{L}_0\} \tag{4.4}$$

Taking $p \equiv A \to B$ and $q \equiv \neg B \to A$ and $r \equiv \neg B \to \neg A$, we find that the tuple of propositions

$$\langle\langle (A \to B) \to (\neg B \to A), (\neg B \to A) \to (\neg B \to \neg A)\rangle, (A \to B) \to (\neg B \to \neg A)\rangle$$

does belong to the set in (4.4).

In example (4.2), the proposition $(A \to B) \to (\neg B \to \neg A)$ is said to be a **direct consequence** of the propostions $(A \to B) \to (\neg B \to A)$ and $(\neg B \to A) \to (\neg B \to \neg A)$ by virtue of the rule (4.3).

Note that an inference rule could be an infinitary relation. All that we want is that membership in an inference rule must be decidable. This is most easily achieved by describing the "shapes" of the premises and the conclusion of the inference rule in terms of the syntactic patterns of the formulas involved in the rule. In fact it is obvious from the form of the rule (4.2) that it describes the shape characteristics of the formulas involved in its premises and conclusion. The same also holds for axiom schemas as in example (4.1).

This point will become clearer when we study proof systems for propositional logic. This is especially true in the case when we need to define an infinite set of axioms for our theory. If this infinite set of axioms can be partitioned into a finite collection of different "syntactic patterns" then decidability of membership reduces to determining whether a given formula can be expressed as a particular instance of one of the given finite set of patterns.

Definition 4.2 *The axioms and rules of inference of a formal theory $\mathcal{T}$ together constitute a* ***proof system*** *for the set of wffs of the theory.*

Definition 4.3 *A formal theory $\mathcal{I}$ is called* **axiomatic** *if there exists an algorithm which can determine for any wff whether or not it is an axiom.*

Hence for a formal theory (with a possibly infinite number of axioms!) to be axiomatic, it is sufficient that there exists a finite collection of patterns or forms by which all the axioms can be represented. In other words, if there are an infinite number of axioms, then they should all be representable using some finite sequences of symbols, so that an algorithm may be employed to determine the "axiom-hood" or otherwise of any arbitrary wff. Therefore in an axiomatic theory it is common to find a finite set of **axiom schemas**, where each schema could represent an infinite number of axioms, all of which have a similar pattern.

Notes

1. Very often we regard axioms (and axiom schemas) as rules of inference with an empty set of premises .
2. The main difference between an axiom and an axiom schema is that whereas an axiom is a single wff, an axiom schema denotes a set (possibly infinite) of wffs all of which have the "shape" defined by the schema.

Definition 4.4 *Let Γ be an arbitrary set of wffs in the lanaguage of a theory $\mathcal{T}$. A* **witnessing deduction** *π is a finite sequence $p_1, p_2, \ldots, p_n$ of wffs such that for each p_i, $1 \leq i \leq n$,*

1. *$p_i \in \Gamma$, or*
2. *p_i is an axiom, or*
3. *p_1 is a direct consequence of some wffs* preceding *it in the sequence, by virtue of a rule of inference. That is p_i is a direct consequence of wffs $p_{k_1}, \ldots, p_{k_m}$ in π by virtue of some m-ary rule of inference such that $k_j < i$ for each k_j, $1 \leq j \leq m$.*

This witnessing deduction is also called a **(formal) proof** *of p_n from Γ and we write $\Gamma \vdash_{\mathcal{T}} p_n$ to denote that there exists a proof of p_n from Γ. The subscript "$_{\mathcal{T}}$" is often omitted when the theory being referred to is understood.*

The above definition of a formal proof captures the following points we already expect correct proofs of a conclusion q from some set of hypotheses represented by Γ, to satisfy.

1. A proof is a sequence of steps.
2. The first step of the proof can only be an assumption from the set of hypotheses or an application of an axiom (schema).

3. Every other step if it is not an assumption or an application of an axiom (schema) must follow from previous steps (and should possess adequate justification!). This ensures that there is no circularity in proofs.

By keeping the definition of a proof so simple, we have not tackled the question of what adequate justification really means. Often in mathematics we use previously proven theorems as justification. We do not actually formalize it here in the concept of a formal proof for the simple reason, that we could always replace such a justification by a copy of the proof of the theorem referred to. This simplicity in the definition of a formal proof also ensures that circularity cannot creep in indirect ways (e.g. the proof of theorem T1 requires theorem T2, while the proof of theorem T2 requires yet another theorem T3 and the proof of T3 requires T1, etc.). In practice, however we will often use the results of previously proven theorems as justification[1]. It is then to be understood as trying to abbreviate the proof for convenience without losing any rigour.

Definition 4.5 *Let $p_1, p_2, \ldots, p_n \equiv q$ be a proof of q from Γ. Construct the following directed graph of at least n nodes, such that*

- *each node represents a wff in the proof*
- *there are directed edges from the node representing p_i in the proof to nodes $p_{k_1}, \ldots, p_{k_m}$ in the graph if and only if p_i is a direct consequence of $p_{k_1}, \ldots, p_{k_m}$, by virtue of some m-ary rule of inference.*
- *Replicate nodes which have more than one edge coming in to ensure that every node has in-degree at most* 1.

The resulting graph is called a **proof tree** *for $\Gamma \vdash q$, rooted at q.*

It should be intuitively clear to the intelligent reader from the above construction that the graph so obtained is indeed a tree.

Questions.

1. How do you know the constructed graph is acyclic?
2. What can you say about the wffs represented by isolated nodes in the dag ?
3. What you say about the wffs at the root of the resulting proof tree ?
4. What you say about the wffs which are the leaves of the resulting proof tree?
5. What can you say about wffs in Γ that do nopt occur in either the proof or the proof tree.

[1] Think of the theorem as the name of a macro whose expansion is its proof.

Definition 4.6 *A* **(formal) theorem** *of a theory $\mathcal{T}$ is a wff q such that there exists a proof of q from an empty set of premises. We write $\vdash_{\mathcal{I}} q$ to denote that q is a theorem in $\mathcal{I}$.*

In other words q is a formal theorem in $\mathcal{T}$ if there exists a sequence $p_1, \ldots, p_n$ such that each p_i $(1 \leq i \leq n)$ is either an axiom or a direct consequence of some of the preceding wffs in the sequence by virtue of a rule of inference. Further p_1 can only be an axiom. Also of necessity every proof is a finite sequence. Even though sometimes infinitary proofs are used in mathematical theories (usually by inductive arguments), the finitary nature of a proof is often taken for granted.

Remarks.

1. Every premise in Γ is a consequence of Γ.
2. The first step in a proof cannot be obtained by virtue of a rule of inference.
3. If q is a theorem then the first step in any proof of q must be an axiom.
4. Every axiom of a formal theory is also a theorem, since there is a proof of length 1 whose first and last step is the axiom.

The facts in the following theorem may be proved by the reader using the definitions we have given so far.

Theorem 4.1 *Let Γ and Δ be sets of wffs in a theory $\mathcal{T}$.*

Monotonicity *If $\Gamma \subseteq \Delta$ and $\Gamma \vdash q$, then $\Delta \vdash q$.*

Compactness *$\Delta \vdash q$ if and only if there is a finite subset $\Gamma \subseteq \Delta$ such that $\Gamma \vdash q$.*

Substitutivity *If $\Delta \vdash q$ and for each $p \in \Delta$, $\Gamma \vdash p$, then $\Gamma \vdash q$.*

Definition 4.7 *A theory is called* ***decidable*** *if there exists an algorithm, which can determine whether a given wff is a theorem or not. Otherwise the theory is said to be* ***undecidable****.*

It is important to note that nowhere in a formal theory does the semantics of the language (in case the wffs of the theory are defined linguistically) play a role. However in constructing a formal theory, we must ensure that the theory is in fact, compeletely reconciled with the semantics of the language. Otherwise the theory is either inconsistent or it is a theory of something else. In the following section we progress towards such a consistent and complete theory for Propositional Logic. Needless to say, it would involve using semantics to justify the axioms and the inference rules of the theory, though the theory itself may be presented without any indication of the semantics of the language that we are considering.

4.3 A Hilbert-style Proof System for Propositional Logic

In this section we propose a formal theory $\mathcal{H}_0$ for Propositional Logic. We choose a minimalist version of both the language and the system of axioms that we need. However, we would not be sacrificing either the expressive power of the language or the set of theorems that may be proven.

In contrast having a rich system may pose several problems. Firstly the system may be inconsistent, in the sense that unless we are very careful, every wff may actually be a theorem of the system. Such a theory would be of little use to man or beast as it would allow us to prove contradictory statements.

Secondly, while a rich system of axioms and inference rules (which is consistent) is very helpful for a human being to construct proofs (it also has the advantages that there may be several different ways of proving the same thing), it is easier to deal with a small but complete set of axioms and inference rules which allow you to prove all provable facts.

The ease of dealing with a small system relates to the ability to show first of all that the system is consistent which often would require a tedious case analysis on the axioms and rules of inference. It may also involve constructing actual models which may be used in real life reasoning. Model construction can be made easier by reducing the system to a minimal one.

Moreover, at a more fundamental level, there is the question of why should there be a large proof system, if a smaller and more compact one can do the same job.

There are several proof systems for propositional logic available, and we will consider only one of them in detail. We denote this particular theory by $\mathcal{H}_0$.

Definition 4.8 *The formal theory $\mathcal{H}_0$ consists of*

1. *The language $\mathbb{L}_0(\neg, \rightarrow)$ inductively defined from a countable set of atoms and having as constructors only the operators $\neg$ and $\rightarrow$. The well-formed formulas of the theory are the elements of $\mathbb{L}_0(\neg, \rightarrow)$.*
2. *There are three axiom schemas named* K*,* S*,* N *and a single rule of inference named* MP *(standing for modus ponens). They are shown below.*

(K)	$\vdash X \rightarrow (Y \rightarrow X)$
(S)	$\vdash (X \rightarrow (Y \rightarrow Z)) \rightarrow ((X \rightarrow Y) \rightarrow (X \rightarrow Z))$
(N)	$\vdash (\neg Y \rightarrow \neg X) \rightarrow ((\neg Y \rightarrow X) \rightarrow Y)$
(MP)	$(X \rightarrow Y), X \vdash Y$

Notational Conventions and Explanatory Comments.

1. We will generally avoid the use of unnecessary parentheses, by assuming that $\neg$ has a higher precedence than $\rightarrow$ and $\rightarrow$ associates to the right. This is largely for convenience and is not part of the formal theory.

2. In order to make clear difference between an axiom and an axiom schema, we use the upper case roman letters 'X', 'Y', 'Z' (possibly decorated with subscripts, superscripts etc.) to denote "place-holders" (or names of places in a context, where arbitrary wffs can be substituted in those places, with the proviso that if any two places have same name then the same wff is substituted in both places).

3. For example, the axiom schema K represents an infinite number of axioms all of which have the same "shape" constraint. That is, K denotes the set of formulas

   $$\{(p \rightarrow (q \rightarrow p)) \mid p, q \in \mathbb{L}_0\}$$

   Similarly the rule MP denotes the relation

   $$\{\langle\langle P \rightarrow q, p\rangle, q\rangle \mid p, q \in L_0\}$$

   Similar comments apply to other axiom schemas.

4. It has become a practice in any rule of inference to write out the premisses simply separated by commas, without using the conventional set braces ("{" and "}").

5. We use the "turnstile" symbol ($\vdash$) to separate the hypotheses from the conclusion in a rule of inference. An axiom schema (or an axiom) is written starting with the "turnstile" to denote the fact that it has an empty set of premises.

Exercises

1. Show that every axiom is a tautology.

2. Show using the truth table technique that MP is a valid inference rule.

3. Design algorithms to determine whether a wff in L_0 is an axiom.

4. Design an algorithm to determine whether for any three wffs p,q,r of L_0 , r is obtained from p and q by virtue of the modus Ponens rule of inference.

In the sequel we will prove various theorems of the propositional logic. As a matter concerning proof methodology we may state here that for the sake of convenience , brevity and to avoid repetition we will often use previously proved theorems in performing deductions. This is perfectly justified by substitutivity and is convenient so as to avoid repetition. Similarly, we might also define and prove new rules of inference from the existing ones and invoke them in further proofs[2].

We will number the theorems of the formal theory of propositional logic either as T1, T2, . . .etc, or in certain cases where there is a well known name we will use the name. Similarly the new rules that we introduce will be numbered R1, R2 . . . etc. in the sequence in which they appear.

We begin with the reflexivity property of the conditional. To aid the reader in understanding proofs, we give the substitutions that we make in the axioms and the rule of inference as part of the justification of every step.

Lemma 4.1 (Reflexivity) $\vdash p \rightarrow p$ *for propostion p.*

Proof: Let p be any wff of $\mathbb{L}_0$.

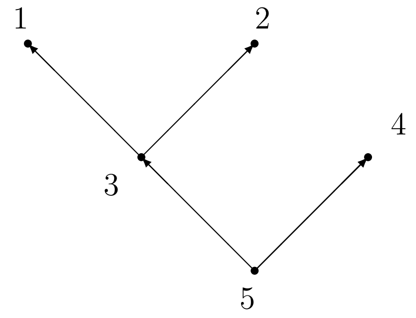
1. $p \rightarrow ((p \rightarrow p) \rightarrow p)$ $\quad \mathsf{K}\{p/X, p \rightarrow p/Y\}$
2. $(p \rightarrow ((p \rightarrow p) \rightarrow p)) \rightarrow ((p \rightarrow (p \rightarrow p)) \rightarrow (p \rightarrow p))$ $\quad \mathsf{S}\{p/X, p \rightarrow p/Y, p/Z\}$
3. $((p \rightarrow (p \rightarrow p)) \rightarrow (p \rightarrow p))$ $\quad \mathsf{MP}(2, 1)$
4. $p \rightarrow (p \rightarrow p)$ $\quad \mathsf{K}\{p/X, p/Y\}$
5. $p \rightarrow p$ $\quad \mathsf{MP}(3, 4)$

□

The following diagram illustrates the notion of a proof tree for lemma (4.3).

For the sake of brevity we have labelled each node by its sequence number in the proof.

[2] An important issue and one which we must guard against is that of circularity i.e. proving a theorem by using another proof depends upon the theorem under consideration . But a careful analysis of whatever we have stated in the foregoing should convince the reader that there is nothing circular in using the result of a previously proved theorem to prove a following theorem



Proof tree for theorem T1(Reflexivity)

Digression.
T1. $\vdash\ X \rightarrow X$ is really a theorem of $\mathcal{I}$ in conformance with the definition of a formal theory. However we may prove certain properties about a formal theory itself. For instances, the fact that we have earlier demonstrated about formal theories are proved not within any formal theory but in an informal style as is done in mathematics. Hence any statement about a formal theory or about theories in general are called **meta-statements** and theorem, lemmata etc. that are necessary are appropriately enough called **meta-theorems**

In most natural language discourses, the distinction is seldom made between a statement and a statement about statements. This is one reason made why reasoning in natural language is prone to confusion. But more importantly, by not making these distinctions, one might be led to inconsistency.The problem of self-reference which led to the paradoxes [3] both in natural

[3] The liar paradox is a simple example. Consider the statement "This statement is false". The question is whether "this statement is false" is true or false. If this statement were true then the statement is obviously false. On the other hand if this statement were false, then the statement is true. Hence the statement is both true and false.

Another simple example is the Grelling paradox. An adjective is said to be auto-logical if the property denoted by the adjective is true for the adjective. Otherwise the adjective is said to be hetero-logical. For example the word "poly-syllabic" is auto-logical because "poly-syllabic" is poly-syllabic whereas "mono-syllabic" is not a "mono-syllabic" word. The question now is whether the adjective "hetero-logical" is heterological or not. We leave it to the reader to prove that it is both hetero-logical and not hetero-logical.

language and in set theory[4] may be attributed directly to this lack of distinction between statements and meta-statements. In fact we are discussing the properties of one language containing English and a large part of mathematics.

End of digression.

Most mathematical theory may be stated in the form "if p then q", for some propositions p and q. Very often the proof of such a theorem begins by assuming that p is true and then goes on to show that q is also true. In other words, whereas the statement of the theorem is of the forms $\vdash p \rightarrow q$ the proof really is a proof of $p \vdash q$. The following theorem and its converse show that the two equivalent (at least in the formal theory $\mathcal{I}_0$).

Meta-theorem(The Deduction Theorem). If $\Gamma, p \vdash q$ then $\Gamma \vdash p \rightarrow q$ for all wffs p, q of L_0 and $\Gamma \subseteq L_0$
Proof. Let the sequence $p_1, p_2, \ldots, p_n \equiv q$ be a proof of $p \rightarrow q$ from Γ. This would follow from the following stronger claim, which we prove by induction.

Claim. $\Gamma \vdash p \rightarrow p_i$ for all $i, 1 \leq i \leq n$
Proof of Claim. By induction on i.
Basis: i=1. by definition p_i must be either a premise or an axiom.
in other words, we have three cases, $p_i \equiv p, p_i \in \Gamma$ or p_i is an axiom.
Cases: $p_i \in \Gamma$ or p_i is an axiom. Then
1. $\Gamma \vdash p_i$ Premise
2. $\Gamma \vdash p_i \rightarrow (p \rightarrow p_i)$ A1, $X \equiv p_i, Y \equiv p$
3. $\Gamma \vdash p \rightarrow p_i$ 1,2,MP
Case: $p_i \equiv p$. The claim follows from reflexivity (T11) and fact 1.

That proves the basis of the induction.

Induction hypothesis : $\Gamma \vdash p \rightarrow p$ *for all k,* $1 \leq k < i \leq n$. |

Now consider p_i. Now we have four cases of which three are as in the basis, viz. $p_i \equiv p$, $p_i \in \Gamma or p_i$ is an axiom. The proof of the claim for all these cases is identical to the proof of the basis.
The last case is treated below.

Case: p_i was obtained from $p_j and p_m$ by virtue of MP, j<i and m<i.
Then without loss of generality we may assume $p_m \equiv p_j \rightarrow p_i$. By induction hypothesis we

[4] For a discussion on Russell's, Cantor's and Burali-Forti's paradoxes, refer to any good text on axiomatic set theory.

know $\Gamma \vdash p \rightarrow p_j$ and $\Gamma \vdash p \rightarrow p_m$, i.e.$\Gamma \vdash p \rightarrow (p_j \rightarrow p_i)$ Consider the proofs in which $p \rightarrow p_j$ and $p \rightarrow (p_j \rightarrow p_i)$ are consequence of Γ. We assume that $p \rightarrow p_j$ was obtained in j' steps and $p \rightarrow (p_j \rightarrow p_i$ was obtained in m' steps. Since both proofs are consequences of some set of the premises we may concatenate them in a larger proof and continue the proof as follows.

$$\begin{array}{rll} j'+m'+1. & (p \rightarrow (p_j \rightarrow p_i)) \rightarrow & \\ & \qquad ((p \rightarrow p_j) \rightarrow (p \rightarrow p_i)) & A2, X \equiv p, Y \equiv p_j, Z \equiv p_i \\ j'+m'+2. & ((p \rightarrow p_j) \rightarrow (p \rightarrow p_i)) & m', j'+m'+1, MP \\ j'+m'+3. & p \rightarrow p_i & j', j'+m'+2, MP \end{array}$$

Hence $\Gamma \vdash p \rightarrow p_i$

End of claim

For i=n, it is clear from the claim that $\Gamma \vdash p \rightarrow p_n$.

Meta-theorem (Converse of the deduction theorem). For all Γ, p, q, if $\Gamma \vdash p \rightarrow q$ then $\Gamma, p \vdash q$.

proof. Assume $\Gamma \vdash p \rightarrow q$ has a proof of length m. We reproduce the same proof with p as an added assumption and extended it as follows.

4.4 A Natural Deduction Proof System

Rules of Natural Deduction

$$(\neg - \mathsf{I}) \quad \frac{\Gamma, \phi \vdash \mathbf{ff}}{\Gamma \vdash \neg\phi} \qquad\qquad (\neg - \mathsf{E}) \quad \frac{\Gamma, \neg\phi \vdash \mathbf{ff}}{\Gamma \vdash \phi}$$

$$(\wedge - \mathsf{I}) \quad \frac{\Gamma \vdash \phi, \quad \Gamma \vdash \psi}{\Gamma \vdash \phi \wedge \psi}$$

$$(\wedge - \mathsf{E1}) \quad \frac{\Gamma \vdash \phi \wedge \psi}{\Gamma \vdash \phi} \qquad\qquad (\wedge - \mathsf{E2}) \quad \frac{\Gamma \vdash \phi \wedge \psi}{\Gamma \vdash \psi}$$

$$(\vee - \mathsf{I1}) \quad \frac{\Gamma \vdash \psi}{\Gamma \vdash \phi \vee \psi} \qquad\qquad (\vee - \mathsf{I2}) \quad \frac{\Gamma \vdash \phi}{\Gamma \vdash \phi \vee \psi}$$

$$(\vee - \mathsf{E}) \quad \frac{\Gamma \vdash \phi \vee \psi, \quad \Gamma \vdash \phi \to \chi, \quad \Gamma \vdash \psi \to \chi}{\Gamma \vdash \chi}$$

$$(\to -\mathsf{I}) \quad \frac{\Gamma, \phi \vdash \psi}{\Gamma \vdash \phi \to \psi} \qquad\qquad (\to -\mathsf{E}) \quad \frac{\Gamma \vdash \phi \to \psi, \quad \Gamma \vdash \phi}{\Gamma \vdash \psi}$$

$$(\leftrightarrow -\mathsf{I}) \quad \frac{\Gamma \vdash \phi \to \psi, \quad \Gamma \vdash \psi \to \phi}{\Gamma \vdash \phi \leftrightarrow \psi}$$

$$(\leftrightarrow -\mathsf{E1}) \quad \frac{\Gamma \vdash \phi \leftrightarrow \psi}{\Gamma \vdash \phi \to \psi} \qquad\qquad (\leftrightarrow -\mathsf{E2}) \quad \frac{\Gamma \vdash \phi \leftrightarrow \psi}{\Gamma \vdash \psi \to \phi}$$

4.5 Natural Deduction proofs of the Hilbert-style axioms

In this section we use the natural deduction system to derive some important formulas and axiom-schemas. In particular we prove one of the deMorgan laws and the axioms of Hilbert-style proof systems.

Example 4.3 *Use the Natural deduction system to prove the formal theorem*

$$\vdash \neg(\phi \wedge \psi) \to (\neg\phi \vee \neg\psi)$$

Solution. Check the proof tree (Fig. 4.1) whose nodes are labelled with step numbers from the proof.

$$
\begin{array}{lll}
1. & \left[\begin{array}{lll}
1. & \neg(\phi\wedge\psi) & AP1 \\
2. & \phi\vee\neg\phi & \textit{Proved in class} \\
3. & \left[\begin{array}{lll} 1. & \neg\phi & AP2 \\ 2. & \neg\phi\vee\neg\psi & 1.3.1, \vee I \end{array}\right] & \\
4. & \neg\phi\to(\neg\phi\vee\neg\psi) & 1.3, \to I \\
5. & \left[\begin{array}{lll}
1. & \phi & AP3 \\
2. & \left[\begin{array}{lll} 1. & \psi & AP4 \\ 2. & \phi\wedge\psi & 1.5.1, 1.5.2.1, \wedge I \\ 3. & (\phi\wedge\psi)\wedge\neg(\phi\wedge\psi) & 1.1, 1.5.2.2, \wedge I \end{array}\right] & \\
3. & \neg\psi & 1.5.2, \neg I \\
4. & \neg\phi\vee\neg\psi & 1.5.3, \vee I
\end{array}\right] & \\
6. & \phi\to(\neg\phi\vee\neg\psi) & 1.5 \to I \\
7. & \neg\phi\vee\neg\psi & 1.2, 1.4, 1.6, \vee E
\end{array}\right] & \\
2. & \neg(\phi\wedge\psi)\to(\neg\phi\vee\neg\psi) & 1, \to I
\end{array}
$$

Example 4.4 *Here we prove the axiom schema* K *of the Hilbert-style proof system using Natural deduction rules.*

$$
\boxed{\begin{array}{ll}
\boxed{\begin{array}{ll} \textit{0.0.1.}\quad p, q\vdash p & AP \end{array}} & \\
\textit{0.1.}\quad p\vdash q\to p & \to -I(1.1)
\end{array}}
$$

$$
\textit{1.}\quad \vdash p\to(q\to p) \qquad\qquad \to -I
$$

Example 4.5 *We prove the axiom-schema* S *of the Hilbert-style proof system.*

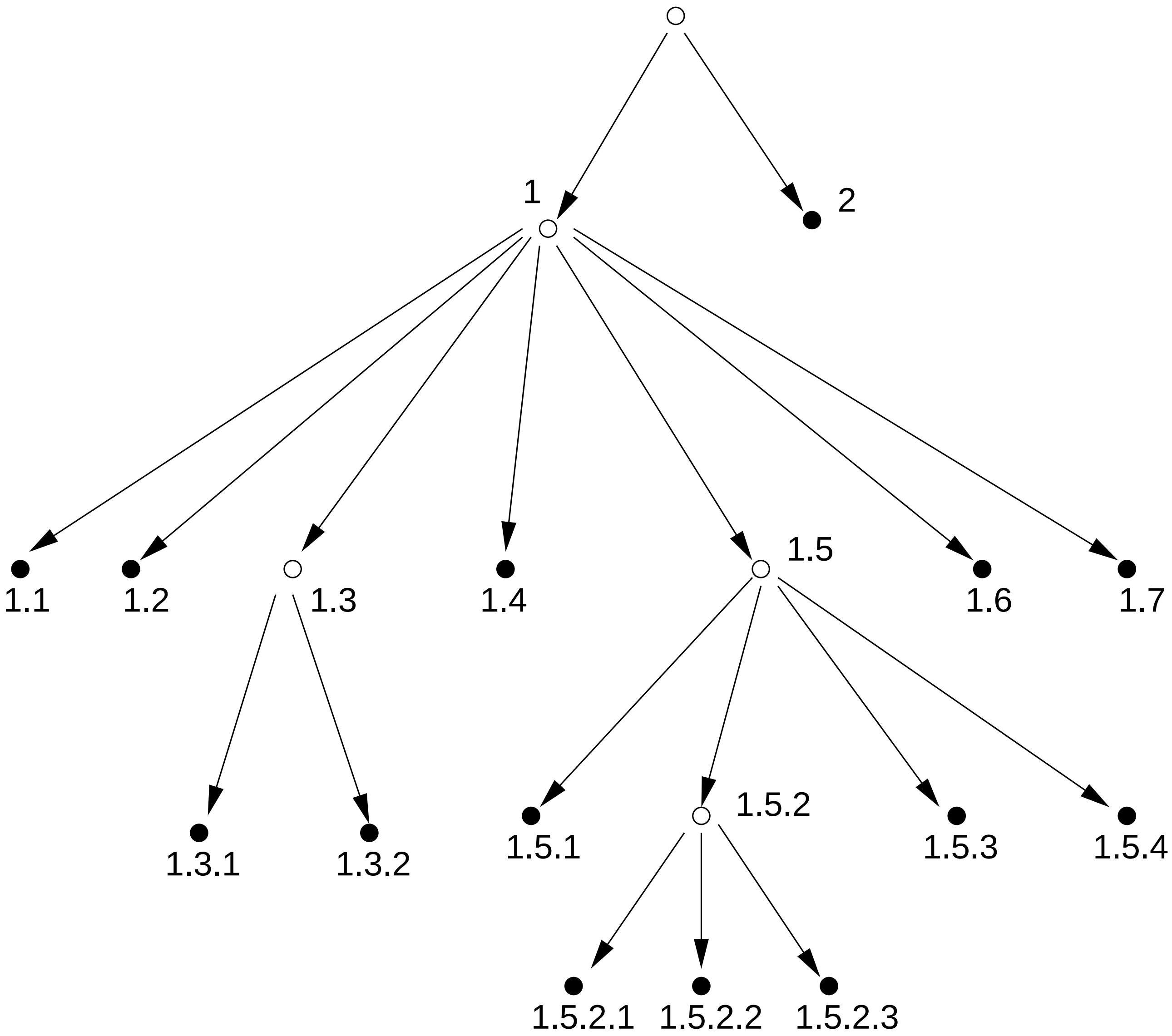


Figure 4.1: The (ecologically sound) proof tree

0.0.0.1.	$p \to (q \to r), p \to q, p \vdash p \to q$	AP
0.0.0.2.	$p \to (q \to r), p \to q, p \vdash p$	AP
0.0.0.3.	$p \to (q \to r), p \to q, p \vdash q$	$\to -E$
0.0.0.4.	$p \to (q \to r), p \to q, p \vdash q \to r$	$\to -E$
0.0.0.5.	$p \to (q \to r), p \to q, p \vdash r$	$\to -E$
0.0.1.	$p \to (q \to r), p \to q \vdash p \to r$	$\to -I$
0.1.	$p \to (q \to r) \vdash (p \to q) \to (p \to r)$	$\to -I$
1.	$\vdash (p \to (q \to r)) \to ((p \to q) \to (p \to r))$	$\to -I$

Example 4.6 *This example gives the proof of the axiom-schema* N *of the Hilbert-style proof system.*

0.0.0.1. $\neg q \rightarrow \neg p, \neg q \rightarrow p \vdash p$

0.0.0.2. $\neg q \rightarrow \neg p, \neg q \rightarrow p \vdash \neg p$

0.0.0.3. $\neg q \rightarrow \neg p, \neg q \rightarrow p \vdash p \wedge \neg p$

0.0.0.4. $\neg q \rightarrow \neg p, \neg q \rightarrow p \vdash \mathbf{ff}$ $\neg - E$

0.0.1. $\neg q \rightarrow \neg p, \neg q \rightarrow p \vdash q$ $\neg - E$

0.1. $\neg q \rightarrow \neg p \vdash (\neg q \rightarrow p) \rightarrow q$ $\rightarrow -I$

1. $\vdash (\neg q \rightarrow \neg p) \rightarrow ((\neg q \rightarrow p) \rightarrow q)$ $\rightarrow -I$

Laws of logical equivalence

To complete the algebraic structure of a Boolean algebra we have introduced two constants $\mathbf{tt}$ and $\mathbf{ff}$ in the language of propositional logic which correspond to the truth values 1 and 0 respectively. But they may be defined as abbreviations of the formulae $\phi \vee \neg\phi$ and $\phi \wedge \neg\phi$ respectively.

$(\phi \vee \psi) \vee \chi \Leftrightarrow \phi \vee (\psi \vee \chi)$	Associativity	$(\phi \wedge \psi) \wedge \chi \Leftrightarrow \phi \wedge (\psi \wedge \chi)$
$\phi \vee \psi \Leftrightarrow \psi \vee \phi$	Commutativity	$\phi \wedge \psi \Leftrightarrow \psi \wedge \phi$
$\phi \vee \phi \Leftrightarrow \phi$	Idempotence	$\phi \wedge \phi \Leftrightarrow \phi$
$\phi \vee \mathbf{ff} \Leftrightarrow \phi$	Identity	$\phi \wedge \mathbf{tt} \Leftrightarrow \phi$
$\phi \vee \mathbf{tt} \Leftrightarrow \mathbf{tt}$	Zero	$\phi \wedge \mathbf{ff} \Leftrightarrow \mathbf{ff}$
$\phi \wedge (\psi \vee \chi) \Leftrightarrow (\phi \wedge \psi) \vee (\phi \wedge \chi)$	Distributivity	$\phi \vee (\psi \wedge \chi) \Leftrightarrow (\phi \vee \psi) \wedge (\phi \vee \chi)$
$\neg(\phi \vee \psi) \Leftrightarrow \neg\phi \wedge \neg\psi$	De Morgan	$\neg(\phi \wedge \psi) \Leftrightarrow \neg\phi \vee \neg\psi$
$\phi \wedge (\neg\phi \vee \psi) \Leftrightarrow \phi \wedge \psi$	Absorption	$\phi \vee (\neg\phi \wedge \psi) \Leftrightarrow \phi \vee \psi$
$\phi \wedge (\phi \vee \psi) \Leftrightarrow \phi$	Adsorption	$\phi \vee (\phi \wedge \psi) \Leftrightarrow \phi$
$\neg\neg\phi \Leftrightarrow \phi$	Negation	$\neg\mathbf{tt} \Leftrightarrow \mathbf{ff}$, $\neg\mathbf{ff} \Leftrightarrow \mathbf{tt}$
$\phi \wedge \neg\phi \Leftrightarrow \mathbf{ff}$	Complements	$\phi \vee \neg\phi \Leftrightarrow \mathbf{tt}$
$\phi \rightarrow \psi \Leftrightarrow \neg\phi \vee \psi$	Conditional	
$\phi \leftrightarrow \psi \Leftrightarrow (\phi \rightarrow \psi) \wedge (\psi \rightarrow \phi)$	Biconditional	$\phi \leftrightarrow \psi \Leftrightarrow (\phi \wedge \psi) \vee (\neg\phi \wedge \neg\psi)$

4.6 Derived Operators and Derived Inference Rules

The Language L_0 consists of only two operators.However they are adequate for the whole of propositional logic. The operators may be defined in terms of these as follows. Within the context of L_0 the other operators are **derived** operators. They are defined as follows.

$$\boxed{\begin{array}{l} p \vee q \equiv (\neg p \rightarrow q) \\ p \wedge q \equiv \neg(p \rightarrow \neg q) \\ p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p) \end{array}}$$

There are several deductively complete systems for logic. Many of them are very small. But they also have the disadvantage that they are not very intuitive and intuitively valid deductions require a great deal of thought and practice in such systems. However , we could always define new **derived** rules of inference which are intuitively appealing to them for future use , so that they are very i complex proofs then become greatly simplified by appealing to them.

Some of these **rules** that are considered necessary are already familiar to the reader from our earlier discussion on validity using the truth table method. Consider the following rules of deduction. It is very easy to see by means of the truth table that they are all valid argument forms. But even more interesting and important is the fact that these rules may be safely added as new theorems in the systems $\overset{\Omega}{\sim}_0$,where the other operators are new derived operators are regarded as derived operators of the language L_0.

We leave the proofs of these derived rules as the exercise for the reader.

Exercises.

1. Prove the following rules of inference in the system $\overset{\Omega}{\sim}_0$.

Simplification.	S1	$X \wedge Y \vdash X$	S2.	$X \wedge Y \vdash Y$
Conjunction.	C1.	$X, Y \vdash X \wedge Y$	C2.	$X, Y \vdash Y \wedge X$
Addition.	A1.	$X \vdash X \vee Y$	A2.	$X \vdash Y \vee X$
Disjunctive Syllogism.	DS1.	$X \vee Y, \neg X \vdash Y$	DS2.	$X \vee Y, \neg Y \vdash X$
Modus Tollens.	MT.	$X \rightarrow Y, \neg Y \vdash \neg X$		
Constructive Dilemma.	CD.	$X \rightarrow Y, Z \rightarrow W,\ X \vee Z \vdash Y \vee W$		

2. Prove using the truth table technique that the above argument forms are all valid.

3. Prove all the logical equivalences given in the last chapter using the proof system of $\overset{\Omega}{\sim}_0$.

We now posses enough ammunition in the form of rules and axioms and theorems to be able to reason correctly about valid arguments without resorting to the truth-table technique. In fact, by using the derived rules given in the last exercise set, one may actually almost the way one would reason with a child in the natural language. We give examples of a few such arguments and their proofs.

Example. Consider the following argument form and its proof which is given without repeating the premisses. The conclusion is separated from the last premiss by a '$\vdash$' and a line separate the premisses from the rest of the proof of the argument.

1. $A \vee \neg C$
2. $C \vee D$
3. $\neg A$ $\qquad \vdash D \vee B$

4.	$\neg C$	1, 3, DS
5.	D	2, 4, DS
6.	$D \vee B$	5, A

Example. Here is another example.

1. $\neg A \rightarrow (C \rightarrow E)$
2. $B \vee D \rightarrow (F \rightarrow (G \rightarrow H))$
3. $A \vee (B \vee D)$
4. $\neg A \wedge (C \vee F)$ $\qquad \vdash E \vee (G \wedge H)$

5.	$\neg A$	4, S
6.	$C \rightarrow E$	1, 5, MP
7.	$B \vee D$	3, 5, DS
8.	$F \rightarrow (G \wedge H)$	2, 7, MP
9.	$C \vee F$	4, S
10.	$E \vee (G \wedge H)$	6, 8, 9, CD

Logical Equivalence and Rule E

As must be clear to any reader who has attempt the exercises of the last chapter there are a large number of equivalence that are tautologies (see question 8 in the exercises of the last chapter). It was also mentioned that $p \leftrightarrow q$ is a tautology exactly when both p and q have the same truth value – that is,in any state they are either both true in the state or they are both false. Therefore as far as questions of validity and deductions are concerned it should be clear that if $p \leftrightarrow q$ is a tautology then the replacement of one by the other will not affect the validity of an argument . In fact we may state that as a theorem.

Metatheorem. If $\vdash p \leftrightarrow q$ and $\Gamma \vdash p$ then $\Gamma \vdash q$.

We leave the proof of this theorem as an exercise and concentrate instead on its application.

Example. Consider the following argument on the existence of God under the assumption that there is evil in the world.

- *If God exists then he is omnipotent and beneficient to all.*
- *If God is omnipotent He is able to prevent Evil.*
- *If God is beneficient to all then He is willing to prevent evil.*
- *If there is Evil, then God is not willing or not able to prevent it.*
- *There is Evil.*
- *Therefore there exists no god.*

We "symbolize" (i.e. give appropriate symbols to identify the various atomic propositions in the argument and then rewrite the premisses and the conclusion using these symbols and the logical connectives) the above argument as follows.

G	*God exists*
O	*God is Omnipotent*
B	*God is Beneficient to all*
W	*God is Willing to prevent evil*
A	*God is Able to prevent evil*
E	*There is Evil*

Argument and its proof are given below:

1.	$G \rightarrow (O \wedge B)$	
2.	$O \rightarrow A$	
3.	$B \rightarrow W$	
4.	$E \rightarrow \neg W \vee \neg A$	
5.	E	$\vdash \neg G$

6.	$\neg W \lor \neg A$	5,4,MP
7.	$\neg(W \land A)$	6,E
8.	$\neg A \rightarrow \neg O$	2,E
9.	$\neg W \rightarrow \neg B$	3,E
10.	$\neg O \lor \neg B$ 9.8.6,CD	
11.	$\neg(O \land B)$	10,E
12.	$\neg G$ 1,11,MT	

In the above example note the steps 7, 8 and 9, where E is given as a justification. A closer look at these steps shows that we have used one of the tautology $(p \rightarrow q) \leftrightarrow (\neg q \rightarrow \neg p)$. In the previous chapter we also emphasized that logical equivalence is in fact, a congruence relation. Hence given any proposition r in which p occurs as a subformula, the truth value of the proposition remains unaffected by systematic replacing all occurrence of p by q. Put algebraically, this is possible because logical equivalence is a congruence relation and hence preserve not truth or validity, it also preserves falsity and inconsistency. The above theorem in fact may be extended to take this property of the logical equivalence to give us a new rule, which we call the equivalence rule.

Meta-Theorem (Rule E). Let r be a wff containing the wff p as a subformula. Let the wff s be obtained from r by replacing one or more occurrences of p in r by q. If $\neg p \leftrightarrow q$ and $\Gamma \neg r$ then $\Gamma \neg s$.

The above meta theorem allows us to do pattern matching and substitution on arbitrary subformulas that may be nested deeply within some formula. This is in direct contrast to the other rules of inference , which permit a pattern matching and substitution only at the top levels of formulas and in order to perform such substitutions deeply, it is usually necessary to extract those subformulas out and then apply the rules of inference. We refer to all application of this special rule of inference by the name E (standing for "equivalence")

In the last example the context in each case where the rule E was applied not on any proper subformula,but to whole formulas. This need not always be the case,since the above meta theorem allows its application even for proper subformulas in a proof.the following example illustrates its use in its complete generality.

Example (Taken from McKay). Consider the following argument of home finances. We symbolize the prepositions in the argument in a manner that should be obvious to the intelligent reader.

1. *If we buy a cd player, we must cut down on small purchases or our financial plans will*

be ruined.

2. *If we buy more compact discs, we will not be cutting down on small purchases.*

3. *But if we buy a CD player, we will buy more compact discs.*

Therefore if we buy a cd player our financial plans will be ruined.

The three premisses and the proof of conclusion are given below. In each line which has been justified by an appeal to the rule E, we have also mentioned, within braces, the form of logical equivalence (that is a tautology) that has been used to perform replacements.

1.	$P \to (C \lor R)$	
2.	$M \to \neg C$	
3.	$P \to M$	$\vdash P \to R$

_ _

4.	$P \to \neg C$	2,3,HS	
5.	$P \to \neg\neg C \lor R$	1,E	$\{p \Leftrightarrow \neg\neg p\ \}$
6.	$P \to (\neg C \to R)$	5,E	$\{\neg p \lor q \Leftrightarrow p \to q\}$
7.	$P \land \neg C \to R$	6,E	$\{p \to (q \to r) \Leftrightarrow p \land q \to r\ \}$
8.	$\neg C \land P \to R$	7,E	{Commutativity of $\land$ }
9.	$\neg C \to (P \to R)$	8,E	$\{p \to (q \to r) \Leftrightarrow p \land q \to r\ \}$
10.	$M \to (P \to R)$	2,9,HS	
11.	$P \to (P \to R)$	3,10,HS	
12.	$P \land P \to R$	11,E	$\{\ p \to (q \to r) \Leftrightarrow p \land q \to r\ \}$
13.	$P \to R$	12,E	{ Idempotence of$\land$ }

In cases where the logical equivalence used are not exercises already solved by the reader we urge him to prove these equivalences in $\overset{\Omega}{\sim}_0$.

Exercise

1. Prove metatheorem 1 stated above.

2. Prove metatheorem 2 stated above (*Hint. Use structural induction)*

More examples on reasoning – Conditional proofs and indirect proofs.

We illustrate in the following examples the use of the deduction theorem in performing conditional proofs, indirect proofs and proofs by cases. A proof which uses the deduction theorem directly is called a **conditional** proof and one which proves statement by proving a contradiction is called an **indirect** proof.

Example. We consider the CD player example which was previously proved by a direct proof and now give a proof which uses the deduction theorem.

1. $P \rightarrow (C \vee R)$
2. $M \rightarrow \neg C$
3. $P \rightarrow M$ $\qquad \vdash P \Rightarrow R$

|	4.	P	Assumption Premiss (AP)
|	5.	M	3,4,MP
|	6	$C \vee R$	1,4,MP
|	7.	$\neg C$	2,5,MP
|	8.	R	6,7,DS
9.	$P \rightarrow R$		4-8,DT

Note that the conditional proof steps 4-8 in the above example form a new **scope**[5] (we indent it to indicate that it is a new scope). It is very important to realise that in any proof which implies the deduction theorem, preservation of truth (which is the hall-mark of a valid argument) crucially depends upon the truth of the new premiss. Hence what is considered truth preserving in the scope of the new premiss may not be true at all in the proof outside scope. Hence none of the steps in the new scope can be referred to outside the boundaries of that acope. It should be intuitively clear that all proofs by contradiction reallly do add an extra premiss and use the deduction theorem to prove that the assumption of the premiss leads to a contradiction i.e. the premiss is actually false.

However, within a given scope we may use steps that lie in an enclosing scope without any fear of voilating the validity of the argument.

The conditional proof methaod and the restrictions we have imposed in the previous paragraph find their echoes everywhere in mathematics. Since most theorems are expressed in the form of conditional sr=tatements, many proofs would begin by assuming the antecedent and show that the consequence follows from the antecedent by appealing to the premisses and to previously

[5] The rules governing the use of scopes are very similar to the rules of reference and usage of variables in statically scoped programming languages like Pascal, ML etc. (as oppose to the dynamically scoped languages like LISP, Scheme etc.)

proved lemmata/theorems and/or postulates/definitions. But no proof of a theorem would be considered correct if it appealed to some step in the proof of another theorem in which an extra assumption has been made (which subsequently leads to a contradiction). We may think of the indented portion in the above example as a proof of the conditional statement immidiately following it. Further we may consider the statement immidiately following a conditaional proof to be a lemma which was not previously proved, but is somehow necessary for the proof of a main theorem. Then the proof of the theorem would obviously appeal only to the lemma and not the steps leading to its proof. Consider the following example which has two conditional nested one within the other.

Example

1.	$A \rightarrow B$	
2.	$C \vee D$	$\vdash A \rightarrow (\neg D \rightarrow (B \wedge C))$

– –

|	3.	A		AP
|	4.	B		1,3,MP
|	|	5.	$\neg D$	AP
|	|	6.	C	2,5, DS
|	|	7.	$B \wedge C$	4,7,C
|	8.	$\neg D \rightarrow B(B \wedge C)$		5-7,DT
9.	$A \rightarrow (\neg D \rightarrow B \wedge C)$			3-8,DT

It is easy to see that as step 10 one cannot conclude $B \wedge C$ from steps 4 and 6 and an appeal to the conjuction rule because in the state in which A, B and c are all false and d is true, the premisses are all true but the conclusion would be false. However *under the assumption* that A is true and d is false (which are APs in steps 3 and 5 respectively), we validly conclude $B \wedge C$ (as can be seen in step 7).

As a matter of general strategy, it usually pays the conditional proof (CP) method whwnever the topmost operator in the conclusion is an implication. The reader will see more examples of the use of CP in the sequel.

Since a proof by contradiction (indirect proof) also uses extra premisses, the same scope rules that apply to conditional proofs also apply in the case of indirect proofs.

Example We prove the argument about the nonexistent of God using the indirect proof method.

1. $G \to (O \wedge B)$
2. $O \to A$
3. $B \to W$
4. $E \to \neg W \vee \neg A$
5. E $\vdash \neg G$

6		$\neg W \vee \neg A$	4,5,MP
|	7.	G	AP
|	8.	$O \wedge B$	1,7,MP
|	9.	O	8,S
|	10.	A	C2,9,MP
|	11.	B	8,S
|	12.	W	3,11,MP
|	13.	$W \wedge A$	12,10,C
|	14.	$\neg(\neg W \vee \neg A)$	13,E
|	15.	$(\neg W \vee \neg A) \wedge \neg(\neg W \vee \neg A)$	6,14,E
16.		$\neg G$	7-15,IP

Proofs by Cases

The proof method that we would like to mention is the method of proof by cases. Normally when facts have to be proved (for instance about the various constructs of programming language) we usually do an exhaustive case analysis and show that the property we have set out to prove holds for each individual case (for a programming language, it holds for each possible different costruct). Such proofs are called proofs by cases is really nothing more than an application of the constructive dilemma rule.

$p \wedge q$
$p \to r$
$q \to r$
$r \vee r$
r

However it is important to realise that using proof by cases involes the use of tautologies in the proof. We explain this throgh the following example.

Example The above proof may also be done by cases and by conditional proofs.

1. $B \to P$
2. $P \to (N \to M)$
3. $\neg B \to (I \wedge T)$ $\vdash N \to (T \vee M)$

4.		$B \vee \neg B$	tautology
|	5.	N :w	AP
|	|	6. B	AP
|	|	7. P	1,6,MP
|	|	8. $N \to M$	2,7, MP
|	|	9. M	5,8, MP
|	|	10. $T \vee M$	9,A
|	11	$B \to T \vee M$	6,10,CP
|	|	12. $\neg B$	AP
|	|	13. $I \wedge T$	3,12,MP
|	|	14. T	13,s
|	|	15. $T \vee M$	14, A
|	16	$\neg B \to T \vee M$	12-15,CP
|	17	$(T \vee M) \vee (T \vee M)$	4.11,16,CD
|	18.	$T \vee M$	17,E
19		$N \to T \vee M$	5-18,CP

The above example is the illustrative of two things. Firstly, one can nest conditional proofs to any level of nesting provided that we make sure that we follow the acope rules in our in our derivations. secondaly note the use of a tautology in step 4. It is easy to realize that the use of tautologies does not affect the validity of a proof. In fact, often a proof by cases method will use the above disjuntive tautology to exhastively list out all cases, so that the CD rule may be evenualy applied.

Exercises.

1. Prove th following valid argument. Prove at least one argument each by direct proof, conditional proof, indirect proof and proof by cases.

   (a) $A \to B, \neg A \to C \vdash (A \wedge B) \vee (\neg A \wedge C)$

   (b) $(A \wedge B) \vee (\neg A \wedge C) \vdash (A \to B) \wedge (\neg A \to C)$

   (c) $B \to (C \wedge \neg D), (C \wedge A) \to \neg E, F \to (D \vee E) \vdash A \to (B \to \neg F)$

   (d) $\neg(A \wedge (B \vee C), D \to B, E \vee C \vdash A \to (E \wedge \neg D)$

   (e) $\neg A \to (B \to C), A \vee B, B \to \neg C, A \to D \vdash A \wedge D$

   (f) $A \to (c \wedge D), B \to (E \vee \neg C), E \to (\neg A \vee \neg D) \vdash \neg(A \wedge B)$

(g) $A \rightarrow (\neg B \wedge \neg C), C \vee E, B \rightarrow (E \wedge \neg D), A \vee B \vdash E$

2. Inconsistency of a set of sentences may be proved using the deduction system by assuminng all the sentences as premisses and then deriving a contradiction. Prove the inconsistency of the following sets.

   (a) $\{A \wedge \neg B, A \rightarrow (B \vee C), \neg C\}$

   (b) $\{\neg A \rightarrow (C \vee D), C \rightarrow D, \neg(D \vee A)\}$

   (c) $\{\neg(B \leftrightarrow C) \rightarrow A, D \rightarrow \neg A, A \rightarrow (D \wedge C),\ B \leftrightarrow \neg c\}$

   (d) $\{S \rightarrow \neg F, \neg(S \vee C) \rightarrow T, \neg(T \vee M), \neg C, F \wedge M\}$

3. You are present in the Congressional hearing on the IRAN CONTRA deal. The following statements were made by REAGAN, BUSH and NORTH (the three accused in the case) respectively in their testimonies:

   REAGAN: BUSH is innocent but NORTH is guilty.
   BUSH: I am innocent but on of the other two are guilty.
   NORTH: If REAGAN is guilty then so is BUSH.

   (a) Symbolize the statement made by the three accused.

   (b) Assuming that all three spoke the truth deduce from the rules of inference, who is guilty and who is innocent.

   (c) If the innocent spoke the truth and the guilty lied deduce who is innocent and who is guilty.

   (d) Use the truth table method to verify that your answers to part b) and c) are indeed correct.

4. Inspector Ghote (of H.R.F. Keating fame) has finally apprehended three suspects ARVE, AHBARVE and CARVE, one of whom has murdered DHARVE. The three suspects made the following statements.

   ARVE: I did not kill Dharve. Dharve was an old acquaintance of BARVE'S and CARVE hated Dharve.

   BARVE: I did not murder Dharve. I did not even no him. Besides, I was out of town the day murder took place.

   CARVE: I did not murder Dharve. I saw both Arve and Barve with Dharve on the of murder; one of them must have done it.

Inspector Ghote assume that the innocent are telling the truth whereas the guilty man may or may not be lying. How did he solve the murder?

5. Consider the following argument.
If the price rise, then the poor and and the salaried class will be unhappy. If taxes are increased then the businessmen will be unhappy. If the poor and the salaries class or the businessmen are unhappy. If the poor and the Government will not be re-elected. Inflation will rise if Government expenditure exceeds its revenue. Government expenditure will exceed ids revenue unless taxes are increased or the Government resorts to deficit financing or takes a loan from the IMF.

   (a) What should the Government do to get re-elected?

   (b) Now add two more premisses:
   If the Government resorts to deficit financing then its expenditure will not exceed its revenue. But if it resorts to deficit financing then prices will rise .
   What should the Government do to re-elected?

   (c) Suppose the following extra premiss is added:
   The Government cannot borrow from the IMF if its expenditure exceeds its revenue.

Can you then prove that the Government will not get re-elected?

m+1. $\Gamma, p \vdash p \rightarrow q$ m,F1.
m+2. $\Gamma, p \vdash p$ Premise
m+3. $\Gamma, p \vdash q$ m+1, m+2, MP.

The deduction theorem and its converse are very useful in proving theorems and convenient derived rules of inference. We number the derived inference rules by either giving them names and by numbering them as 'R?'.

Exercises.

1. Prove the following theorems and derived inference rules of $\overset{\omega}{\sim}_0$.
   R1. $X \rightarrow Y, Y \rightarrow Z \vdash X \rightarrow Z$
   R2. $X \rightarrow (Y \rightarrow Z), Y \vdash X \rightarrow Z$
   T2. $\vdash \neg\neg X \rightarrow X$
   T3. $\vdash X \rightarrow \neg\neg X$
   T4. $\vdash \neg X \rightarrow (X \rightarrow Y)$
   T5. $\vdash (\neg Y \rightarrow \neg X) \rightarrow (X \rightarrow Y)$
   T6. $\vdash (X \rightarrow Y) \rightarrow (\neg Y \rightarrow \neg X)$
   T7. $\vdash X \rightarrow (\neg Y \rightarrow \neg(X \rightarrow Y))$
   T8. $\vdash (X \rightarrow Y) \rightarrow ((\neg X \rightarrow Y) \rightarrow Y)$

2. Other important proof techniques used in most mathematical theories are those of *proof by contradiction* (also called the *indirect proof method*) and *proof by cases.* These methods may be stated as derived rules of inference respectively. Prove the following metatheorems.

   (a) *Indirect Proof Method.* $\Gamma \vdash X$ if and only if $\Gamma, \neg X \vdash \neg(Y \rightarrow Y)$

   (b) *Proof by Cases.* $\Gamma \vdash X$ if and only if $\Gamma, \neg Y \rightarrow z \vdash Y \rightarrow X$ and $\Gamma, \neg Y \rightarrow z \vdash Z \rightarrow X$

4.7 Consistency, completeness and decidability

There are several important issues to be dealt with n any mathematical theory. Foremost among these is the question of whether the axioms of the theory are consistent. In other words, any theory is quite useless if contradictory statements can be proved in the theory. Naturally, if a contradiction and hence is a theorem.

Secondly, given a formal theory, we have some notion of what the statements in the theory mean to us. In the case of propositional logic, we have defined a semantics. However in the formal theory, we have not mention the semantics anywhere. Yet, we would like to relate the semantics with the proof theory. Viewed from one angle, it is meant to check that the proof theory is consistent with the semantics. Looked at another way, we would also like to determine what facts in the theory are provable from the proof theory. Although in the case of propositional logic, truth table are adequate to give us all the valid arguments we require, we will see that in first order logic, truth tables may be of infinite size and therefore can not be constructed at all. In such a situation we have to fall back on the proof theory to get whatever theorems e require.

Given that the proof theory may be the only feasible tool available to us, we would like to be able to determine to what extent is possible to derive the facts that are implied by the semantics. This is the notion of completeness of a theory. A further question is, to what extent can the theory be automated to give us answers to question we might ask of it. In other words is the theory (or some large part of it) decidable? In this section we try to answer these questions by first relating the semantics to the proof theory.

Definition. A formal theory is called **consistent** if not all the wffs are theorems.

We could have of course,defined a formal theory as being consistent if it is not possible to derive a contradiction. However, such a definition would not be applicable in a formal theory in which there is no (explicit) negation and it would become very difficult to show that it is consistent. In the case of any theory of propositional logic which has '$\neg$' as an operator however, the two definitions would be equivalent.

Note that all axioms of a theory are in facts theorems (since there is a proof of length l, from the empty set of premisses).

Exercise.

1. Prove that for some wff p in L_0, if both p and $\neg$p are theorems of $\overset{\Omega}{\sim}_0$ then every wff in L_0. is a theorem of $\overset{\Omega}{\sim}_0$.

Metatheorem. Every theorem of $\overset{\Omega}{\sim}_0$ is a tautology.
Proof. We prove this by induction on the lengths of proofs.
Base. Proofs of length 1. The claim is trivial since the first step of the proof of the any theorem

must be an axiom. But this first step is also the theorem. Since we know from the exercises that each axiom is a tautology, it follows that all theorems with proofs of length 1 are tautologies. *Induction hypothesis : Every theorem of L_0 which has a proof of length m, $0 \leq m \leq n$, is a tautology.*

Let q be any theorem of $\overset{\Omega}{\sim}_0$ whose proof is of length n. Let $p_1, \ldots, p_n \equiv q$ be the sequence of wffs which constitute the proof of q. If q is an axiom then q was obtained by virtue of R_0 on some preceding wffs p_i and p_j , such that $i \leq n$ and $j \leq N$. without loss of generality we may assume that $p_j \equiv p_i \rightarrow q$. By the induction hypothesis, we know that p_i and p_j are tautologies and is easy to prove the following claim from the semantics.

Claim. if p and $p \rightarrow q$ are both tautologies then q is a tautology.
It follows therefore that q is a tautology.

Corollary. $\overset{\Omega}{\sim}_0$ is consistent. *Proof.* Since the only theorems of $\overset{\Omega}{\sim}_0$ are tautologies and there exist wffs such as $\neg(p \rightarrow p)$ which are not tautologies, it follows that $\overset{\Omega}{\sim}_0$ is consistent.

Before we show the completeness we need to prove a meta-lemma which shows that corresponding to every row of the truth table of a proposition there exists a deduction in the proof system. Equivalently, for every state there exists a deduction involving only the atoms in a proposition or their negations. In the completeness theorem we will use this fact.

Example. Consider any wff, say $\neg(P \rightarrow \neg Q)$. Its truth table is shown below.

Now consider any arbitrary row of the truth table say the third row where P is true and Q is false. In this row we see that $\neg(P \rightarrow \neg Q)$ is false. The deduction may be obtained from the proof system as follows. (For convenience , we also list out the premisses in each step).

1.	$\neg Q, P \vdash Q$	Premiss
2.	$\neg Q, \vdash P \rightarrow \neg Q$	1,deduction theorem
3.	$\neg Q, P \vdash P \rightarrow \neg Q$	2, F1
4.	$\neg Q, P \vdash (P \rightarrow \neg Q) \rightarrow \neg\neg(P \rightarrow \neg Q)$	T3
5.	$\neg Q, P \vdash \neg\neg(P \rightarrow \neg Q)$	4,3,MP

Note that the state information was used only to set up the deduction. Other rows of the truth table gives rise to other deductions corresponding to rows 1,2 and 4 are respectively, $\neg Q$, $\neg P \vdash \neg\neg(P \rightarrow \neg Q), Q$, $\neg P \vdash \neg\neg(P \rightarrow \neg Q)$ and Q, $P \vdash \neg(P \rightarrow \neg Q)$. The reader is encouraged to proved these from the proof system. In other words, corresponding to each state there exists a deduction. The form of the deduction should by now be clear to the reader.

Metalemma. Let p be a wff containing the atoms $p_1, \dots, p_k$. For each state σ and all i, $1 \leq i \leq k$, let

$$P_i^* \equiv \begin{cases} P_i & if \quad \sigma(P_i) = 1 \\ \neg P_i & if \quad \sigma(P_i) = 0 \end{cases}$$

and

$$P^* \equiv \begin{cases} p & if \quad \mathcal{T}[P_i]_\sigma = 1 \\ \neg p & if \quad \mathcal{T}[P_i]_\sigma = 1 \end{cases}$$

Then $P_1^*, \dots, P_k^* \vdash p^*$.

Proof By induction on the number of occurrences of the operators in p. Let n be the number of occurrences of operators in p.
Base case n=0. Clearly then p is an atom (p $\equiv P_1$) and hence is made up of a single atom. it is clear that if $\sigma(P_1) = 1$ then $p^* \equiv P_1^* \equiv P_1$ and $P_1 \vdash P_1$. Similarly if $\sigma(P_1) = 0$ then $p^* \equiv P_1^* \equiv \neg P_1$ and $\neg P_1 \vdash \neg P_1$.
Induction hypothesis. The claim holds for all wffs with less than n ($n \leq 0$) occurrence of the operators.

Suppose p is a wff with n operators. Then there are two cases to consider.
Case $p \equiv \neg q$, where q has less than n operators. Then by the induction hypothesis we have $P_1^*, \dots, P_k^* \vdash q^*$.

Case $\mathcal{T}[q]_\sigma = 1$. Then $\mathcal{T}[p]_\sigma = 0$. and $q^* \equiv q$ and $p^* \equiv \neg p \equiv \neg\neg q$. Then we have the following deduction.

1. $P_1^*, \dots, P_k^* \vdash q$ induction hypothesis
2. $P_1^*, \dots, P_k^* \vdash \rightarrow \neg\neg q$ T3 and F1
3. $P_1^*, \dots, P_k^* \vdash \neg\neg q$ 1,2,MP

Hence $P_1^*, \dots, P_k^* \vdash p$.

Case $\mathcal{T}[q]_\sigma = 0$. Then $\mathcal{T}[p]_\sigma = 1$. and $q^* \equiv \neg q$ and $p^* \equiv p \equiv \neg q$. Then we have $P_1^*, \dots, P_k^* \vdash \neg q$ by the induction hypothesis and hence $P_1^*, \dots, P_k^* \vdash p^*$.

Case $p \equiv q \rightarrow r$, where q and r have less than n operators. Then by the induction hypothesis we have $P_1^*, \dots, P_k^* \vdash q^*$. $P_1^*, \dots, P_k^* \vdash r^*$. There are three cases to consider which are treated in order.

Case $\mathcal{T}[q]_\sigma = 0$. Then $\mathcal{T}[p]_\sigma = 1$. and $q^* \equiv \neg q$ and $p^* \equiv p \equiv q \rightarrow r$. We then have the following deduction.

1. $P_1^*, \dots, P_k^* \vdash \neg q$ induction hypothesis
2. $P_1^*, \dots, P_k^* \vdash \neg q \rightarrow (q \rightarrow r)$ T4
3. $P_1^*, \dots, P_k^* \vdash p^*$ 1,2,MP

Case $\mathcal{T}[r]_\sigma = 1$. Then $\mathcal{T}[p]_\sigma = 1$. and $r^* \equiv r$ and $p^* \equiv p \equiv q \rightarrow r$. We then have the following deduction.

1.	$P_1^*, \ldots, P_k^* \vdash r$	induction hypothesis
2.	$P_1^*, \ldots, P_k^* \vdash r \rightarrow (q \rightarrow r)$	A1 and F1
3.	$P_1^*, \ldots, P_k^* \vdash q \rightarrow r$	1,2,MP

Hence $P_1^*, \ldots, P_k^* \vdash p$.

Case $\mathcal{T}[q]_\sigma = 1.$ and $\mathcal{T}[q]_\sigma = 1.$ Then $\mathcal{T}[p]_\sigma = 0$, $q^* \equiv q$,$r^* \equiv \neg r$ and $p^* \equiv \neg p \equiv \neg(q \rightarrow r)$. We then have the following deduction.

1.	$P_1^*, \ldots, P_k^* \vdash \neg q$	induction hypothesis
2.	$P_1^*, \ldots, P_k^* \vdash \neg r$	induction hypothesis
3.	$P_1^*, \ldots, P_k^* \vdash q \rightarrow (\neg r \rightarrow \neg(q \rightarrow ra))$	T7 and F1
4.	$P_1^*, \ldots, P_k^* \vdash \neg r \rightarrow \neg(q \rightarrow r)$	1,3,MP
5.	$P_1^*, \ldots, P_k^* \vdash \neg(q \rightarrow r)$	1,3,MP

Hence $P_1^*, \ldots, P_k^* \vdash p$.

It is important to realize that all we have done is to draw up a correspondence between the 2^k rows of a truth table with 2^k different deductions that involve any wff p containing k atoms. these deductions are proofs in the strict proof-theoretic sense, but each of them represents a row of the truth table. The important link that this establishes is that each row of a truth table of size 2^k could equally well be represented by a set of k premisses, where each premiss is an atom if the truth assignment is 0. Further we have shown that from these premisses, the wff p may be proven if it is true in state and its negation may be proven if it false in the state.

The fact that the proof allows us to represent states and the truth values of compound propositions gives us intuitive evidence that the theory $\overset{\Omega}{\sim}$ is quite rich. the previous lemma shows that the truth values of every compound proposition in every state may be obtained from the proof theory by a deduction from a set of premisses which represent the relevant truth assignments in that state. So it might seem since every theorem of $\overset{\Omega}{\sim}_0$ is a tautology (and therefore its truth value is independent of states), it should be possible to get rid of the premiss somehow. That is indeed what we do in the following theorem.

Metatheorem (completeness of $\overset{\Omega}{\sim}_0$). Every tautology of L_0 is a theorem of $\overset{\Omega}{\sim}_0$.
Proof. Let p be any tautology made of the atoms $P_1, \ldots, P_k$. In any state σ, $p^* \equiv p$.
Claim. $P_1^*, \ldots, P_{k-1}^* \vdash p$.
Proof of claim. Consider any two rows of the truth table which differ only in the truth value assigned to P_k and are identical otherwise. By the previous metalemma we have the following two deductions corresponding to the rows of the truth table in which $\sigma(P_k) = 1$ and $\sigma(P_k) = 0$ respectively.
$P_1^*, \ldots, P_{k-1}^*, P_k \vdash p$
$P_1^*, \ldots, P_{k-1}^*, \neg P_k \vdash p$

Note that the premisses differ only for k. By the deduction theorem we can find deductions
$P_1^*, \dots, P_{k-1}^*, \vdash P_k \vdash p$
$P_1^*, \dots, P_{k-1}^*, \vdash \neg P_k \vdash p$
corresponding to the rows respectively. Since the premisses in the two deductions are identical they may be concatenated to obtained the following deductions (assuming that the conditions are lengths i and i' respectively).

1.	$P_1^* \dots, P_{k-1}^* \vdash \dots$	
.	$\dots$	
.	$\dots$	
.	$\dots$	
i.	$P_1^* \dots, P_{k-1}^* \vdash P_k \to p$	
i+1.	$P_1^* \dots, P_{k-1}^* \vdash \dots$	
.	$\dots$	
.	$\dots$	
.	$\dots$	
i+i'.	$P_1^* \dots, P_{k-1}^* \vdash \neg P_k \to p$	
i+i'+1.	$P_1^* \dots, P_{k-1}^* \vdash (P_k \to p) \to ((\neg P_k \to p) \to p)$	T8,F1
i+i'+2.	$P_1^* \dots, P_{k-1}^* \vdash (\neg P_k \to p)$	i,i+i'+1,MP
i+i'+3.	$P_1^* \dots, P_{k-1}^* \vdash p$	i+i',i+i'+2,MP

End of claim

If k=1 then the claim above is the required proof. If $k \geq 1$ then apply the method used in the claim another k-1 times to eliminate the other premisses. It follows that p is theorem.

The claim in the proof of the above theorem eliminates a premiss by constructing a single deduction for each pair of deductions which differ only in the premiss that is to be eliminated. In effect the claim produces 2^{k-1} different deductions from the original 2^k by coalescing these pairs appropriately. By repeated application of the claim the new 2^{k-1} deductions may be reduced to a single deduction which has no premisses.

Corollary. $\overset{\Omega}{\sim}_0$ is a decidable theory.
Proof. Since every theorem of $\overset{\Omega}{\sim}_0$ is a tautology and vice versa and there exists an algorithm whether a given wff in L_0 is a tautology, it follows that it is possible to determine by an algorithm whether a given wff is a theorem.

Notes

1. It is important to note that even though we have argued for the decidability of $\overset{\Omega}{\sim}_0$ from the fact that there is an algorithm to decide whether a given wff is a tautology , actually

the proof of the completeness theorem and the metalemma preceding it itself may be regarded as an effective methodor an algorithmfor constructing the proof of the theorem in L_0. even though the method outlined in the proofs is likely to produce very long proof, it is still a valid proof within the meaning of the word in the formal theory.

2. A consequence of the completeness theorem is that every valid argument in L_0 can be proved using the axioms and the modus ponens rule of inference . completeness result is not restricted to only to tautologies but extends also to valid arguments because an arguments of the forms $p_1, \ldots, P_n \vdash q$ is valid if and only if $p_1 \rightarrow (\ldots(p_n \rightarrow q)\ldots)$ is a tautology. Hence from a proof of $p_1 \rightarrow (\ldots(p_n \vdash q$ we may derive $p_1, \ldots, p_n \vdash q$ by n applications of the deduction theorem.

Independence of the axioms of $\overset{\Omega}{\sim}_0$.

We have treated the most important concepts of a formal theory viz. , consistency, completeness and decidability. Once we know that a theory is consistent we can go about trying to various theorems knowing fully well that contradiction can not be proved. Further with completeness, we also know that our proof theory is powerful enough to prove the sort of the facts that we have we are interested in proving about the theory. However, there is always the nagging question of whether we can do with less in sense that will be made clear in the sequel.

Given a set of axioms of any mathematical theory, mathematicians are always interested in knowing whether some of the axioms are actually redundant. That is if we were to remove one or more of these axioms, is the completeness of the theory affected in any way? Equivalently, can one or more of the axioms of the theory be proven from the others ? If so, by removing these axioms we would still have a complete theory.

The idea of obtaining a minimal set of axioms dates back to the middle ages, when several able and not-so-able mathematicians felt that Euclid's parallel postulates[6] was redundant and that it could be proved from the other axioms and postulated of the Euclidean geometry. While nobody doubted the truth that a postulates ought to be. However all attempts to prove it from the other postulates failed[7] . The fact of the matter was simply that the parallel postulate is independent of the other postulates. It is independent in the sense that neither it nor its negation can be deduced from the other postulates. The independence of the postulates was proved much later.

Coming back to the independence of axioms A1-a3 for the theory $\overset{\Omega}{\sim}_0$, we have the following definitions.

[6] Through a given point not on a given line there exists a unique line parallel to the given line.

[7] However one mathematician (Geralamo Sccheri (1667-1773)), who felt it was indeed self -evident, replaced the parallel postulates by assumptions that contradicted it and hoped to arrive at a logical contradiction by a reduction ad absurdum method. He proved many theorem that he regarded as absurd because they did not conform to well understood intuitions concerning euclidean geometry. But there where no logical contradictions anywhere in his work. In fact, he came up with what were the first non-Euclidean geometries , even though he was under the impression that he had successfully proved Euclid right

Definition. Given an axiomatic theory *mathcalI* the set of axioms is said to be **independent** if each axiom of $\mathcal{I}$ is independent of the others. An axiom p of $\mathcal{I}$ is said to be **independent of others** if there exists no proof of p which uses only the other axioms and the rules of inference.

In order to prove that A1-A3 is independent we require to show that each axiom is independent of the others. Assume that we would like to prove that A1 is independent of others . For this purpose it suffices to be able to identify a characteristic property that is satisfied by A2 and A3 and is preserved by the modus ponens rule of inference. We refer to such characteristic properties as hereditary. Then given any proof which does not use A1, we are guaranteed that every step of the proof has this characteristic property if the assumptions in the proof have it.

Definition. A property of the wffs of a formal theory $\mathcal{I}$ is said to be **hereditary** if whenever there is a set of wffs which possess this property, every wff that is deduced from one or more of them by virtue of the rules of inference is guaranteed to possess the property.

We have already encountered such a property.

Example. For example, *truth* (i.e. the property of a proposition being true instead of false) is a characteristic property of all valid arguments and is preserved by all the rules of inference (provided the premisses are true) and is therefore a hereditary property.
Example. *falsity* is not a hereditary property of valid arguments.
Example. The property of a wff being a tautology is a hereditary property, since modus ponens preserves tautologies.

But we can not use truth or ”tautologousness” as the characteristic property for proving the independence of any axiom in $\overset{\Omega}{\sim}_0$. simply because, all the axioms and the inference rule share the property. So it does not clearly differentiate one axiom from the others and the rule of inference. And since *falsity* is not hereditary we can not use that either.

To define a hereditary property, it is also possible to use functions from wffs to some finite set of values, such that the function is defined inductively and its result is a value from the range. The meaning function $\mathcal{L}$ that we have defined earlier is one such with a two-element range representing truth and falsity respectively. We could take some range and define a function f on wffs inductively and then look for a property such as $f(p) = c$ for some constant c in the range of f. Alternatively if the range is some ordered set with a given ordering $\leq$ we could determine whether $f(p) \leq c$ is a hereditary property. We shall follow such an approach.

Hence if we want to prove that the axiom A1 is independent of the other axioms we require to inductively define a function f on wffs and choose a value c from the range such that the following conditions satisfied.

- There exists an instance p of A1, such that $f(p)f(p) = / = c$

- All instances of A2 and A3 satisfy the property that for any instance p, f(p) = c.
- The modus ponens rule of inference preserves this property, i.e. whenever the premisses of the rule satisfy the property f(p)=c, the conclusion does too.

It is then clear that f(p)=c is a hereditary property that will be satisfied by all proofs in which there is no instance of axiom A1. Since A1 does not satisfy this characteristic property, it would follow that A1 is independent of others. To prove the independence of the other axioms we would have to find other such functions with similar properties and prove their independence separately.

In defining these functions, we follow the method of truth tables, except that it may be impossible to find a suitable function with a two element range and define the function through the table.

Metalemma The axioms schema A1 in $\overset{\Omega}{\sim}_0$ is independent of A2 and A3.
Proof consider the function f defined inductively on the formation rules of L_0 by the following tables. The range of f is the set {0,1,2} . The following claims are then easy to check and prove. We say that a wff p is select if f(p) = 0 regardless of the values of its individual atoms.

Claim 1. All instances of A2 and A3 are select.

Claim 2. The modus ponens rule of inference preserves *selectness* , i.e. for all wffs p,q if $f(p \rightarrow q) = f(p) = 0 \ then \ f(q) = 0$.

Claim 3. There are instances of A1 which are not select.

Claim 3 follows from the fact for f(p)=0 and f(q)=1, we have $(p \rightarrow (q \rightarrow p)) = 2$.

p	$\neg p$
0	1
1	1
2	0

p	q	$p \rightarrow q$
0	0	0
0	1	2
0	2	2
1	0	2
1	1	2
1	2	0
2	0	0
2	1	0
2	2	0

The function f

Metalemma The axioms schema A2 in $\overset{\Omega}{\sim}_0$ is independent of A1 and A2.
Proof consider the function f defined inductively on the formation rules of L_0 by the following tables. The range of f is the set {0,1,2} . The following claims are then easy to check and prove. We say that a wff p is select if g(p) = 0 regardless of the values of its individual atoms.

claim 1. All instances of A1 and A3 are *grotesque*.

Claim 2. The modus ponens rule of inference preserves *grotesqueness* i.e. for all wffs p,q if $g(p \rightarrow q) = g(p) = 0theng(q) = 0$.

Claim 3. There are instances of A2 which are not *grotesque*.

p	$\neg p$
0	1
1	0
2	1

p	q	$p \rightarrow q$
0	0	0
0	1	2
0	2	1
1	0	0
1	1	2
1	2	0
2	0	0
2	1	0
2	2	0

The function g

Exercises.

1.define instances of A2 which are not grotesque.
2. Define a similar function h and a hereditary property which will prove that A3 is independent of A1 and A2.

4.8 Compactness

In the completness theorem we have assumed willy-nilly that the set of hypotheses Γ from which a proposition ψ may be proven (using a proof system such as Natural deduction) is finite. Is it possible that there are formulas ψ which are a logical consequence of only an infinite set of assumptions but are not a consequence of any finite set?

In other words, it is instructive to know what happens to the relation $\Gamma \models \psi$ when Γ is an infinite set of assumptions.

The following lemma is quite obvious. Also note from the definition of satisfiability that empty set of formulas is satisfied by every truth assignment.

Lemma 4.2 *If a set* Γ *of formulas is satisfiable then every finite subset of* Γ *is also satisfiable.*

Lemma 4.3 *If a set* Γ *of formulas is unsatisfiable then there exists a finite subset of* Γ *which is also unsatisfiable.*

Proof: Suppose Γ is a set of formulas that is unsatisfiable. Since Γ could be infinite[8] the set $atoms(\Gamma) = \bigcup_{\phi \in \Gamma} atoms(\phi)$ could also be infinite (though each individual set $atoms(\phi)$ is always finite). Let $\{A_0, A_1, A_2, \ldots\}$ be an enumeration of the set $atoms(\Gamma)$.

The construction of the truth assignment tree. Consider the following *finitely branching* but infinite depth binary tree. The tree, starting from a root labelled A_0 has two, branches the left branch signifies that A_0 has been assigned the truth value 0 and the right branch denotes the assignment 1. Each node at depth i, $i \geq 0$ is similarly constructed. This tree is an infinite depth tree with each node having exactly two branches. Note that there an uncountable number of paths in this tree and each path corresponds to exactly one possible truth assignment.

Pruning the tree Since Γ is unsatisfiable, for every possible truth assignment τ, there exists at least only formula $\phi \in \Gamma$ which is false. Since any such formula ϕ is made up of only a finite number of atoms, there exists a finite point in the path corresponding to τ at which it is known that ϕ is false along that path (in fact, ϕ is false along every path on the subtree rooted at that point).

Along each path consider the shortest prefix of that path at which it is known that some formula of Γ is false. Now consider the tree obtained from the original tree by pruning every path at the earliest point at which some formula is ϕ is known to be false.

The finite subset of Γ In the resulting tree every path is of finite length and each path could be labelled with the formula from Γ that it falsifies. Since the tree is finitely branching and every path is of finite length, it follows from Konig's lemma that the tree has only a finite number of nodes. Hence it has only a finite number of paths, with each path labelled by a formula from

[8] We assume that Γ is not uncountable, since the language $\mathcal{P}_0$ has only countably many formulas.

Γ that it falsifies. The set of all such formula is a finite subset $\Delta \subseteq_f \Gamma$. It follows from our construction that Δ is unsatisfiable. □

By combining the above two lemmas, we obtain the compactness theorem, stated below.

Theorem 4.2 *A set Γ of formulas is satisfiable if and only if every finite subset of Γ is satisfiable*

Proof: ($\Rightarrow$). If Γ is satisfiable then there exists a truth assignment τ such that $\mathcal{T}[\![\phi]\!]_\tau = 1$ for each $\phi \in \Gamma$. Clearly every finite subset $\Delta \subseteq_f \Gamma$ is satisfied by τ.

($\Leftarrow$). If Γ is not satisfiable, then we know there exists a finite subset of Γ that is unsatisfiable. But since every finite subset of Γ is satisfiable, Γ is also satisfiable. □

Having proved the compactness theorem, we are now in a position, to answer the questions raised in this section.

Corollary 4.1 $\Gamma \models \psi$ *if and only if there exists a finite subset* $\Delta \subseteq_f \Gamma$ *such that* $\Delta \models \psi$.

Proof: ($\Leftarrow$). Clearly if $\Delta \subseteq_f \Gamma$ such that $\Delta \models \psi$ then $\Gamma \models \psi$.

($\Rightarrow$). Assume $\Gamma \models \psi$. Then we know $\Gamma \cup \{\neg\psi\}$ is unsatisfiable. By compactness there exists a finite subset $\Delta \subseteq_f \Gamma \cup \{\neg\psi\}$ which is also unsatisfiable.

Claim 1: $\neg\psi \in \Delta$.
Proof of claim 1. Suppose $\neg\psi \notin \Delta$ Then $\Delta \subseteq_f \Gamma$. But by compactness then Γ is also unsatisfiable, which is a contradiction. Hence $\neg\psi \in \Delta$.

Claim 2: If $\Delta - \{\neg\psi\}$ is unsatisfiable then $\Gamma \models \psi$.
Proof of claim 2. Suppose $\Delta - \{\neg\psi\}$ is unsatisfiable. Then since $\Delta - \{\neg\psi\} \subseteq_f \Gamma$ it is clear from compactness that Γ is also unsatisfiable. Hence $\Gamma \models \psi$ trivially follows.

Claim 3: If $\Delta - \{\neg\psi\}$ is satisfiable then $\Gamma \models \psi$.
Proof of claim 3. We know that both $\Gamma \cup \{\neg\psi\}$ and Δ are unsatisfiable. Now consider any truth assignment τ. If τ satisfies $\Delta - \{\neg\psi\}$ but not Δ then every formula in $\Delta - \{\neg\psi\}$ must be true under τ and $\neg\psi$ must be false. But this implies ψ must be true under τ.

Hence for every truth assignment τ, $\Delta - \{\neg\psi\} \models \psi$. Since $\Delta - \{\neg\psi\} \subseteq_f \Gamma$, it is clear that $\Gamma \models \psi$. □

We may combine the above corollary with the soundness and completeness results that we have already proved to obtain the following theorem.

Theorem 4.3 *For any set of formulas Γ, $\Gamma \models \psi$ if and only if there exists a finite $\Delta \subseteq_f \Gamma$ such that $\Delta \vdash \psi$.*

Proof: (⇒). Assume $\Gamma \models \psi$, then there exists a finite $\Delta \subseteq_f \Gamma$ such that $\Delta \models \psi$. By the completeness it follows that $\Delta \vdash \psi$.

(⇐). Assume there exists a finite $\Delta \subseteq_f \Gamma$ such that $\Delta \vdash \psi$. Then by the soundness theorem $\Delta \models \psi$, from which it follows trivially that $\Gamma \models \psi$. □

4.9 Propositional Resolution

Chapter 5

Resolution in Propositional Logic

5.1 Introduction

Let Γ be a finite set of propositional formulas and suppose it is necessary to prove for some propositional formula q that $\Gamma \models q$. If we were to attempt to prove this by resolution then we would have to show that $\Gamma \cup \{\neg q\}$ is a contradiction. In other words, we would be showing that $\bigwedge \Gamma \wedge \neg q$ is always false. Let p be the conjunctive-normal form (CNF) of $\bigwedge \Gamma \wedge \neg q$. Let n be the number of distinct atoms occurring in p and let Δ be the list of clauses in Δ. If each clause is also represented as a list then Δ is a *list of lists of literals* where each literal is either an *atom* or the *negation of an atom.*

Note:

1. Mathematically, the only difference between sets and lists is that unlike sets,
   (a) lists are ordered sequences of elements, and
   (b) a list may contain duplicate occurrences of elements.
2. We may assume that there is a predefined *total ordering* of atoms, which is extended to the representation of clauses and lists of clauses, in such a way that every clause made of a set of literals has a *unique* representation as a list of literals sorted according to the total ordering. The ordering may further be lifted to present a unique representation of a list of clauses.
3. The *empty list of clauses* represents a *tautology.*
4. A *nonempty list of clauses*, containing the *empty clause* as an element represents a *contradiction.*

5.2 The Resolution procedure

INPUT: A list of lists of literals.

OUTPUT: A list of lists of literals such that *either*

1. the clause represented by the empty list [] belongs to the list, in which case the list is logically equivalent to the proposition **ff**, *or*
2. it is impossible to perform any more steps of resolution. It is easy to check whether it is possible to perform any more resolution, by determining whether in the *cleaned up* list of clauses, there is any occurrence of a complementary pair. The resulting list of clauses is a logical consequence of the original list Δ and yields a valuation under which $\Gamma \cup \{\neg q\}$ is *not* a contradiction.

ALGORITHM: Throughout the algorithm, we may use set notation and terminology whenever it causes no confusion. In such a case it is automatically implied that we mean the set containing the same elements as the list with the same name. Before we proceed to the main body of the algorithm we define two procedures.

procedure **cleanup**

For any list Φ of clauses,

1. For all clauses C, C', if $C \subseteq C'$, then C' may be deleted from Φ without affecting logical equivalence.
2. Any clause containing *complementary pairs* of literals, may be deleted from Φ without affecting logical equivalence.
3. From any clause, duplicate occurrences of a literal may be deleted without affecting logical equivalence.

The resulting list is said to be **clean** and is logically equivalent to the original list Φ.

procedure **resolution**

For any *clean* list Φ and atom A let $\Lambda, \bar{\Lambda} \subset \Phi$ be the sets such that $A \in C$ for every $C \in \Lambda$ and $\neg A \in \bar{C}$ for each $\bar{C} \in \bar{\Lambda}$. Since Φ is clean the two sets Λ and $\bar{\Lambda}$ satisfy the following properties:

1. $\Lambda \cap \bar{\Lambda} = \emptyset$, i.e. Λ and $\bar{\Lambda}$ are disjoint,

2. for each $C \in \Lambda$ and $\bar{C} \in \bar{\Lambda}$, $A \notin \bar{C}$ and $\neg A \notin C$.

Then the new list Ψ of clauses after resolution is obtained as follows

$$\Psi = (\Phi - (\Lambda \cup \bar{\Lambda})) \cup \{D \mid D = (C - \{A\}) \cup (\bar{C} - \{\neg A\})\}$$

Algorithm *main:*

```
Cleanup $\Delta$;
while ($[] \notin \Delta$) $\wedge$ (a complementary pair $(A, \neg A)$ exists in $\Delta$) do

begin
        perform resolution on $(A, \neg A)$ in $\Delta$, yielding a new list of clauses $\Phi$;
        Cleanup $\Phi$ to yield a new list $\Psi$;
        $\Delta := \Psi$
end.
```

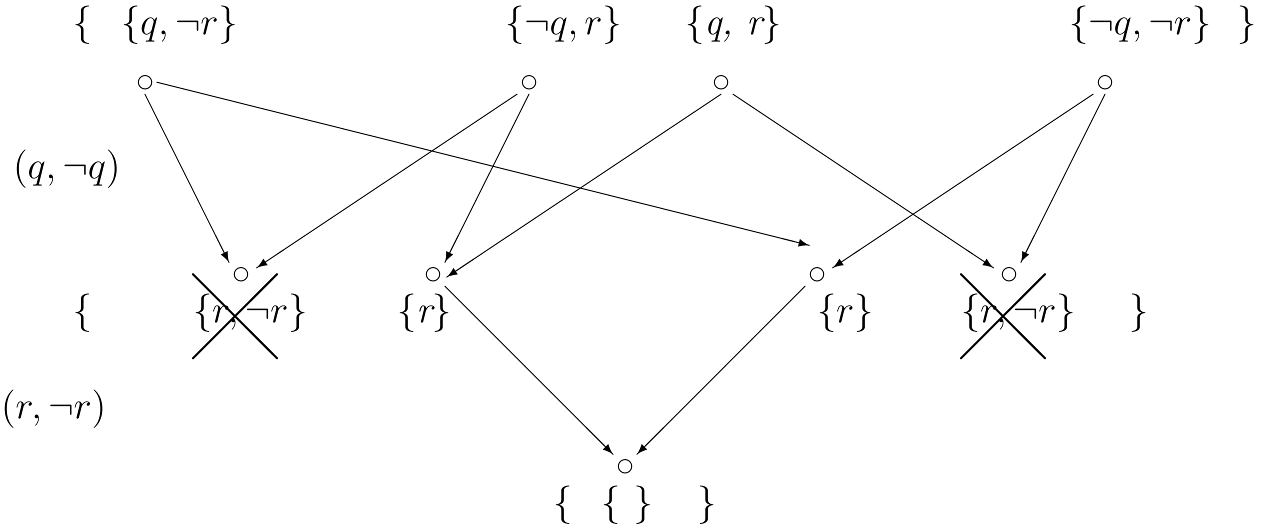


Figure 5.1: A resolution proof

Note that cleaning up the set of clauses is absolutely essential for the algorithm to work correctly. The figure 5.1 shows a resolution proof of the set of clauses corresponding to the contradiction $(q \vee \neg r) \wedge (\neg q \vee r) \wedge (q \vee r) \wedge (\neg q \vee \neg r)$. Try to perform resolution without cleaning up after the first iteration (i.e. don't delete the clause corresponding to $r \vee \neg r$ and see what happens).

5.3 Space Complexity of Propositional Resolution.

Assume n is the number of atoms of which Δ is made up. After some iterations of **resolution** and **cleanup**, assume there are k distinct atoms in the set of clauses on which **resolution** is applied. After performing the **cleanup** procedure there could be *at worst* $\left(\begin{array}{c} 2k \\ k \end{array} \right)$ clauses with each clause containing at most k literals. Now,

$$\left(\begin{array}{c} 2k \\ k \end{array} \right) = \frac{2k(2k-1)\dots(k+1)}{k(k-1)\dots 1}$$

Since for each i, $0 \leq i < k$, $2k - i \geq 2 \times (k - i)$, we get $\left(\begin{array}{c} 2k \\ k \end{array} \right) \geq 2^k$.

For any complementary pair $(A, \neg A)$, it is possible that half of the $\left(\begin{array}{c} 2k \\ k \end{array} \right)$ clauses contain A and the other half contain $\neg A$. This would be the worst possible scenario, as it yields the maximum number of new clauses. In performing a single step of resolution over all possible pairs of clauses, a maximum of $(\frac{1}{2}\left(\begin{array}{c} 2k \\ k \end{array} \right))^2$ different clauses each containing at most $k - 1$ literals are thus obtained. Hence, before applying the **cleanup** procedure the space required could be as high as

$$\begin{array}{rcll} S(k) & = & (k-1)(\frac{1}{2}\left(\begin{array}{c} 2k \\ k \end{array} \right))^2 & \text{atoms} \\ & \geq & \frac{k-1}{4} \times (2^k)^2 & \text{atoms} \\ & = & (k-1) \times 2^{2k-2} & \text{atoms} \end{array}$$

However, after **cleanup** the space requirement reduces to $\left(\begin{array}{c} 2k-2 \\ k-1 \end{array} \right)$ clauses each of size at most $k - 1$ atoms thereby requiring a space of $(k-1)\left(\begin{array}{c} 2k-2 \\ k-1 \end{array} \right)$. Since $k \leq n$ the maximum space required after the first application of resolution and before cleaning up exceeds the space required for all other iterations and could be as high as $(n-1) \times 2^{2n-2}$.

5.4 Time Complexity of Propositional Resolution

Given a space of $k\left(\begin{array}{c} 2k \\ k \end{array} \right)$ to represent the clauses containing at most k atoms, we require a time proportional to this amount of space in order to identify which clauses have to be resolved against a particular complementary pair. After **resolution** we create a space of $(k-1) \times 2^{2k-2}$ which has to be scanned for the **cleanup** operations. Hence the amount of time required to perform a step of resolution and the amount of time required to perform the cleanup are both proportional to $(k-1) \times 2^{2k-2}$. Hence the total time required for performing **resolution** followed by **cleanup** in n iterations (which is the maximum possible)

$$T(n) \quad \geq \quad \textstyle\sum_{k=1}^{n}(k-1) \times 2^{2k-2}$$

which is clearly exponential.

Hence both the *worst case* time and space complexities are exponential in the number of atoms.

The bottom-line however is,

It still works!