Chapter 1

Mathematical Reasoning, Proof Principles and Logic

1.1 Introduction

Mathematicians write proof; most of us write proofs. This leads to the question: Which principles of reasoning do we use when we write proofs?

The goal of this Chapter is to try answering this question. We do so by formalizing the basic rules of reasoning that we use, most of the time unconsciously, in a certain kind of formalism known as a *natural deduction system*. We give a (very) quick introduction to *mathematical logic*, with a very deliberate *proof-theoretic* bent, that is, neglecting almost completely all semantic notions, except at a very intuitive level. We still feel that this approach is fruitful because the mechanical and rules-of-the-game flavor of proof systems is much more easily grasped than semantic concepts. In this approach, we follow Peter Andrew's motto [1]:

"To truth through proof".

We present various natural deduction systems due to Prawitz and Gentzen (in more modern notation), both in their intuitionistic and classical version. The adoption of natural deduction systems as proof systems makes it easy to question the validity of some of the inference rules, such as the *principle of proof by contradiction*. In brief, we try to explain to our readers the difference between *constructive* and *classical* (i.e., not necessarily constructive) proofs. In this respect, we plant the seed that there is a deep relationship between *constructive proofs* and the notion of *computation* (the "Curry-Howard isomorphism" or "formulae–as–types principle", see Section 1.9 and Howard [30]).

1.2 Inference Rules, Deductions, The Proof Systems $\mathcal{N}_m^{\Rightarrow}$ and $\mathcal{N}\mathcal{G}_m^{\Rightarrow}$

In this section, we review some basic proof principles and attempt to clarify, at least informally, what constitutes a mathematical proof.

In order to define the notion of proof rigorously, we would have to define a formal language in which to express statements very precisely and we would have to set up a proof system in terms of axioms and proof rules (also called inference rules). We will not go into this; this would take too much time and besides, this belongs to a logic course, which is not what CIS260 is! Instead, we will content ourselves with an intuitive idea of what a statement is and focus on stating as precisely as possible the rules of logic that are used in constructing proofs. Readers who really want to see a thorough (and rigorous) introduction to logic are referred to Gallier [19] van Dalen [44] or Huth and Ryan [31], a nice text with a Computer Science flavor. A beautiful exposition of logic (from a proof-theoretic point of view) is also given in Troelstra and Schwichtenberg [43], but at a more advanced level. Frank Pfenning has also written an excellent and more extensive introduction to constructive logic. This is available on the web at

http://www.andrew.cmu.edu/course/15-317/handouts/logic.pdf

You should also be aware of CIS482, a very exciting course about logic and its applications in Computer Science. By the way, my book has been out of print for some time but you can get it free (as pdf files) from my logic web site

http://www.cis.upenn.edu/˜jean/gbooks/logic.html

In mathematics, we **prove statements.** Statements may be *atomic* or *compound*, that is, built up from simpler statements using *logical connectives*, such as, *implication* (if–then), *conjunction* (and), *disjunction* (or), *negation* (not) and (existential or universal) *quantifiers*.

As examples of atomic statements, we have:

1. “a student is eager to learn”.
2. “a students wants an A”.
3. “an odd integer is never 0”
4. “the product of two odd integers is odd”

Atomic statements may also contain “variables” (standing for abitrary objects). For example

1. human(x): “x is a human”
2. needs-to-drink(x): “x” needs to drink

An example of a compound statement is

$$\text{human}(x) \Rightarrow \text{needs-to-drink}(x).$$

In the above statement, $\Rightarrow$ is the symbol used for logical implication. If we want to assert that every human needs to drink, we can write

$$\forall x(\text{human}(x) \Rightarrow \text{needs-to-drink}(x));$$

This is read: “for every x, if x is a human then x needs to drink”.

If we want to assert that some human needs to drink we write

$$\exists x(\text{human}(x) \Rightarrow \text{needs-to-drink}(x));$$

This is read: “for some x, if x is a human then x needs to drink”.

We often denote statements (also called *propositions* or *(logical) formulae*) using letters, such as A, B, P, Q, *etc.*, typically upper-case letters (but sometimes greek letters, φ, ψ, *etc.*).

If P and Q are statements, then their *conjunction* is denoted $P \wedge Q$ (say: P and Q), their *disjunction* denoted $P \vee Q$ (say: P or Q), their *implication* $P \Rightarrow Q$ or $P \supset Q$ (say: if P then Q). Some authors use the symbol $\rightarrow$ and write an implication as $P \rightarrow Q$. We do not like to use this notation because the symbol $\rightarrow$ is already used in the notation for functions ($f\colon A \rightarrow B$). We will mostly use the symbol $\Rightarrow$.

We also have the atomic statements $\perp$ (*falsity*), which corresponds to **false** (think of it as the statement which is false no matter what), and the atomic statement $\top$ (*truth*), which corresponds to **true** (think of it as the statement which is always true). The constant $\perp$ is also called *falsum* or *absurdum*. Then, it is convenient to define the *negation* of P as $P \Rightarrow \perp$ and to abbreviate it as $\neg P$ (or sometimes $\sim P$). Thus, $\neg P$ (say: not P) is just a shorthand for $P \Rightarrow \perp$.

Whenever necessary to avoid ambiguities, we add matching parentheses: $(P \wedge Q)$, $(P \vee Q)$, $(P \Rightarrow Q)$. For example, $P \vee Q \wedge R$ is ambigous; it means either $(P \vee (Q \wedge R))$ or $((P \vee Q) \wedge R)$.

Another important logical operator is *equivalence*. If P and Q are statements, then their *equivalence*, denoted $P \equiv Q$ (or $P \Longleftrightarrow Q$), is an abbreviation for $(P \Rightarrow Q) \wedge (Q \Rightarrow P)$. We often say “$P$ if and only if Q” or even “P iff Q” for $P \equiv Q$. As we will see shortly, to prove a logical equivalence, $P \equiv Q$, we have to prove **both** implications $P \Rightarrow Q$ and $Q \Rightarrow P$.

An implication $P \Rightarrow Q$ should be understood as an if–then statement, that is, if P is true then Q is also true. So, the meaning of negation is that if $\neg P$ holds then P must be false. Otherwise, as $\neg P$ is really $P \Rightarrow \perp$, if P were true, then $\perp$ would have to be true, but this is absurd.

Of course, there are problems with the above paragraph. What does truth have to do with all this? What do we mean when we say “P is true”? What is the relationship between truth and provability?

These are actually deep (and tricky!) questions whose answers are not so obvious. One of the major roles of logic is to clarify the notion of truth and its relationship to provability. We will avoid these fundamental issues by dealing exclusively with the notion of proof. So, the big question is: What is a proof?

Typically, the statements that we prove depend on some set of *hypotheses*, also called *premises* (or *assumptions*). As we shall see shortly, this amounts to proving implications of the form

$$(P_1 \land P_2 \land \cdots \land P_n) \Rightarrow Q.$$

However, there are certain advantages in defining the notion of *proof* (or *deduction*) of a proposition from a set of premises. Sets of premises are usually denoted using upper-case greek letters such as Γ or Δ.

Roughly speaking, a *deduction* of a proposition Q from a set of premises Γ is a finite labeled tree whose root is labeled with Q (the *conclusion*), whose leaves are labeled with premises from Γ (possibly with multiple occurrences), and such that every interior node corresponds to a given set of *proof rules* (or *inference rules*). Certain simple deduction trees are declared as obvious proofs, also called *axioms*.

There are many kinds of proofs systems: Hilbert-style systems, Natural-deduction systems, Gentzen sequents systems, *etc.* We describe a so-called *natural-deduction system* invented by G. Gentzen in the early 1930's (and thoroughly investigated by D. Prawitz in the mid 1960's). The major advantage of this system is that it captures quite nicely the "natural" rules of reasoning that one uses when proving mathematical statements. This does not mean that it is easy to find proofs in such a system or that this system is indeed very intuitive! We begin with the inference rules for implication.

In the definition below, the expression Γ, P stands for the union of Γ and P. So, P may already belong to Γ. A picture such as

$$\frac{\Delta}{P}$$

represents a deduction tree whose root is labeled with P and whose leaves are labeled with propositions from Δ (possibly with multiples occurrences). Some of the propositions in Δ may be tagged be variables. The list of untagged propositions in Δ is the list of *premises* of the deduction tree. For example, in the deduction tree below,

$$\dfrac{\dfrac{P \Rightarrow (R \Rightarrow S) \qquad P}{R \Rightarrow S} \qquad \dfrac{Q \Rightarrow R \qquad \dfrac{P \Rightarrow Q \qquad P}{Q}}{R}}{S}$$

no leaf is tagged, so the premises form the set

$$\Delta = \{P \Rightarrow (R \Rightarrow S), P, Q \Rightarrow R, P \Rightarrow Q\},$$

with two occurrences of P, and the conclusion is S.

Certain inferences rules have the effect that some of the original premises may be discarded; the traditional jargon is that some premises may be *discharged* (or *closed*). This this the case for the inference rule whose conclusion is an implication. When one or several occurrences of some proposition, P, are discharged by an inference rule, these occurrences (which label some leaves) are tagged with some new variable not already appearing in the deduction tree. If x is a new tag, the tagged occurrences of P are denoted P^x and we indicate the fact that premises were discharged by that inference by writing x immediately to the right of the inference bar. For example,

$$\cfrac{\cfrac{P^x, Q}{Q}}{P \Rightarrow Q}\; x$$

is a deduction tree in which the premise P is discharged by the inference rule. This deduction tree only has Q as a premise, since P is discharged.

What is the meaning of the horizontal bars? Actually, nothing really! Here, we are victims of an old habit in logic. Observe that there is always a single proposition immediately under a bar but there may be several propositions immediately above a bar. The intended meaning of the bar is that the proposition below it is obtained as the result of applying an inference rule to the propositions above it. For example, in

$$\frac{Q \Rightarrow R \qquad Q}{R}$$

the proposition R is the result of applying the ⇒-elimination rule (see Definition 1.2.1 below) to the two premises $Q \Rightarrow R$ and Q. Thus, the use of the bar is just a convention used by logicians going back at least to the 1900's. Removing the bar everywhere would not change anything to our trees, except perhaps reduce their readability! Since most logic books draw proof trees using bars to indicate inferences, we also use bars in depicting our proof trees.

Since propositions do not arise from the vacuum but instead are built up from a set of atomic propositions using logical connectives (here, ⇒), we assume the existence of an "official set of atomic propositions", $\mathbf{PS} = \{\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3, \cdots\}$. So, for example, $\mathbf{P}_1 \Rightarrow \mathbf{P}_2$ and $\mathbf{P}_1 \Rightarrow (\mathbf{P}_2 \Rightarrow \mathbf{P}_1)$ are propositions. Typically, we will use upper-case letters such as P, Q, R, S, A, B, C, *etc.*, to denote arbitrary propositions formed using atoms from $\mathbf{PS}$.

Definition 1.2.1 The axioms and inference rules for *implicational logic* are:

$$\frac{\Gamma, P}{P}$$

The above is a concise way of denoting a tree whose leaves are labeled with P and the propositions in Γ, each of these proposition (including P) having possibly multiple occurrences but at least one, and whose root is labeled with P. A more explicit form is

$$\frac{\overbrace{P_1, \cdots, P_1}^{k_1}, \cdots, \overbrace{P_i, \cdots, P_i}^{k_i}, \cdots, \overbrace{P_n, \cdots, P_n}^{k_n}}{P_i}$$

where $k_1, \ldots, k_n \geq 0$, $n \geq 1$ and $k_i \geq 1$ for some i with $1 \leq i \leq n$. This axiom says that we always have a deduction of P_i from any set of premises including P_i. Some (or all) of the occurrences of the premises $P_1, \ldots, P_n$ may be tagged with distinct variables.

The $\Rightarrow$-*introduction rule*:

$$\frac{\dfrac{\Gamma, P^x}{Q}}{P \Rightarrow Q}\; x$$

This inference rule says that if there is a deduction of Q from the premises in Γ and from the premise P, then there is a deduction of $P \Rightarrow Q$ from Γ. Note that this inference rule has the additional effect of discharging some occurrences of the premise P. These occurrences are tagged with a new variable, x, and the tag x is also placed immediately to the right of the inference bar. This is a reminder that the deduction tree whose conclusion is $P \Rightarrow Q$ no longer has the occurrences of P labeled with x as premises.

The $\Rightarrow$-*elimination rule*:

$$\frac{\dfrac{\Gamma}{P \Rightarrow Q} \qquad \dfrac{\Delta}{P}}{Q}$$

This rule is also known as *modus ponens*.

In the above axioms and rules, Γ or Δ may be empty and P, Q denote arbitrary propositions built up from the atoms in **PS**. A *deduction tree* is a tree whose interior nodes correspond to applications of the above inference rules. A *proof tree* is a deduction tree such that *all its premises are discharged*. The above proof system is denoted $\mathcal{N}_m^{\Rightarrow}$ (here, the subscript m stands for *minimal*, referring to the fact that this a bare-bone logical system).

In words, the $\Rightarrow$-introduction rule says that in order to prove an implication $P \Rightarrow Q$ from a set of premises Γ, we assume that P has already been proved, add P to the premises in Γ and then prove Q from Γ and P. Once this is done, the premise P is deleted. This rule formalizes the kind of reasoning that we all perform whenever we prove an implication statement. In that sense, it is a natural and familiar rule, except that we perhaps never stopped to think about what we are really doing. However, the business about discharging the premise P when we are through with our argument is a bit puzzling. Most people probably never carry out this “discharge step” consciously, but such a process does take place implicitly.

It might help to view the action of proving an implication $P \Rightarrow Q$ as the construction of a program that converts a proof of P into a proof of Q. Then, if we supply a proof of P as input to this program (the proof of $P \Rightarrow Q$), it will output a proof of Q. So, if we don’t give the right kind of input to this program, for example, a “wrong proof” of P, we should not expect that the program return a proof of Q. However, this does not say that the program is incorrect; the program was designed to do the right thing only if it is given the right kind of input. From this functional point of view (also called, constructive), if we take the simplistic view that P and Q assume the truth values **true** and **false**, we should not be shocked that if we give as input the value **false** (for P), then the truth value of the whole implication $P \Rightarrow Q$ is **true**. The program $P \Rightarrow Q$ is designed to produce the output value **true** (for Q) if it is given the input value **true** (for P). So, this program only goes wrong when, given the input **true** (for P), it returns the value **false** (for Q). In this erroneous case, $P \Rightarrow Q$ should indeed receive the value **false**. However, in all other cases, the program works correctly, even if it is given the wrong input (**false** for P).

1. Only the leaves of a deduction tree may be discharged. Interior nodes, including the root, are *never* discharged.

2. Once a set of leaves labeled with some premise P marked with the label x has been discharged, none of these leaves can be discharged again. So, each label (say x) can only be used once. This corresponds to the fact that some leaves of our deduction trees get “killed off” (discharged).

3. A proof is a deduction tree whose leaves are *all discharged* (Γ is empty). This corresponds to the philosophy that if a proposition has been proved, then the validity of the proof should not depend on any assumptions that are still active. We may think of a deduction tree as an unfinished proof tree.

4. When constructing a proof tree, we have to be careful not to include (accidently) extra premises that end up not beeing discharged. If this happens, we probably made a mistake and the redundant premises should be deleted. On the other hand, if we have a proof tree, we can always add extra premises to the leaves and create a new proof tree from the previous one by discharging all the new premises.

5. Beware, when we deduce that an implication $P \Rightarrow Q$ is provable, we **do not** prove that P **and** Q are provable; we only prove that **if** P is provable **then** Q is provable.

The $\Rightarrow$-elimination rule formalizes the use of *auxiliary lemmas*, a mechanism that we use all the time in making mathematical proofs. Think of $P \Rightarrow Q$ as a lemma that has already been established and belongs to some data base of (useful) lemmas. This lemma says if I can prove P then I can prove Q. Now, suppose that we manage to give a proof of P. It follows from the $\Rightarrow$-elimination rule that Q is also provable.

Observe that in an introduction rule, the conclusion contains the logical connective associated with the rule, in this case, $\Rightarrow$; this justifies the terminology "introduction". On the other hand, in an elimination rule, the logical connective associated with the rule is gone (although it may still appear in Q). The other inference rules for $\wedge$, $\vee$, *etc.*, will follow this pattern of introduction and elimination.

Examples of proof trees.

(a)

$$\cfrac{\cfrac{P^x}{P}}{P \Rightarrow P}\; x$$

So, $P \Rightarrow P$ is provable; this is the least we should expect from our proof system!

(b)

$$\cfrac{\cfrac{\cfrac{\cfrac{(Q \Rightarrow R)^y \qquad \cfrac{(P \Rightarrow Q)^z \qquad P^x}{Q}}{R}}{P \Rightarrow R}\; x}{(Q \Rightarrow R) \Rightarrow (P \Rightarrow R)}\; y}{(P \Rightarrow Q) \Rightarrow ((Q \Rightarrow R) \Rightarrow (P \Rightarrow R))}\; z$$

In order to better appreciate the difference between a deduction tree and a proof tree, consider the following two examples:

1. The tree below is a deduction tree, since two its leaves are labeled with the premises $P \Rightarrow Q$ and $Q \Rightarrow R$, that have not been discharged yet. So, this tree represents a deduction of $P \Rightarrow R$ from the set of premises $\Gamma = \{P \Rightarrow Q, Q \Rightarrow R\}$ but it is *not a proof tree* since $\Gamma \neq \emptyset$. However, observe that the original premise, P, labeled x, has been discharged.

$$\cfrac{\cfrac{Q \Rightarrow R \qquad \cfrac{P \Rightarrow Q \qquad P^x}{Q}}{R}}{P \Rightarrow R}\; x$$

2. The next tree was obtained from the previous one by applying the ⇒-introduction rule which triggered the discharge of the premise $Q \Rightarrow R$ labeled y, which is no longer active. However, the premise $P \Rightarrow Q$ is still active (has not been discharged, yet), so the tree below is a deduction tree of $(Q \Rightarrow R) \Rightarrow (P \Rightarrow R)$ from the set of premises $\Gamma = \{P \Rightarrow Q\}$. It is not yet a proof tree since $\Gamma \neq \emptyset$.

$$\dfrac{\dfrac{\dfrac{(Q \Rightarrow R)^y \qquad \dfrac{P \Rightarrow Q \qquad P^x}{Q}}{R}}{P \Rightarrow R}\,x}{(Q \Rightarrow R) \Rightarrow (P \Rightarrow R)}\,y$$

Finally, one more application of the ⇒-introduction rule will discharged the premise $P \Rightarrow Q$, at last, yielding the proof tree in (b).

(c) In the next example, the two occurrences of A labeled x are discharged simultaneously.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{(A \Rightarrow (B \Rightarrow C))^z \qquad A^x}{B \Rightarrow C} \qquad \dfrac{(A \Rightarrow B)^y \qquad A^x}{B}}{C}}{A \Rightarrow C}\,x}{(A \Rightarrow B) \Rightarrow (A \Rightarrow C)}\,y}{\big(A \Rightarrow (B \Rightarrow C)\big) \Rightarrow \big((A \Rightarrow B) \Rightarrow (A \Rightarrow C)\big)}\,z$$

(d) In contrast to Example (c), in the proof tree below the two occurrences of A are discharged separately. To this effect, they are labeled differently.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{(A \Rightarrow (B \Rightarrow C))^z \qquad A^x}{B \Rightarrow C} \qquad \dfrac{(A \Rightarrow B)^y \qquad A^t}{B}}{C}}{A \Rightarrow C}\,x}{(A \Rightarrow B) \Rightarrow (A \Rightarrow C)}\,y}{\big(A \Rightarrow (B \Rightarrow C)\big) \Rightarrow \big((A \Rightarrow B) \Rightarrow (A \Rightarrow C)\big)}\,z}{A \Rightarrow \Big(\big(A \Rightarrow (B \Rightarrow C)\big) \Rightarrow \big((A \Rightarrow B) \Rightarrow (A \Rightarrow C)\big)\Big)}\,t$$

Remark: How do we find these proof trees? Well, we could try to enumerate all possible proof trees systematically and see if a proof of the desired conclusion turns up. Obviously, this is a very inefficient procedure and moreover, how do we know that all possible proof trees will be generated and how do we know that such a method will terminate after a finite number of steps (what if the proposition proposed as a conclusion of a proof is not provable)? This is a very difficult problem and, in general, it can be shown that there is **no** procedure that will give an answer in all cases and terminate in a finite number of steps for all possible input propositions. We will come back to this point in Section 1.9. However, for the system $\mathcal{N}_m^{\Rightarrow}$, such a procedure exists, but it is not easy to prove that it terminates in all cases and in fact, it can take a very long time.

What we did, and we strongly advise our readers to try it when they attempt to construct proof trees, is to construct the proof tree from the bottom-up, starting from the proposition labeling the root, rather than top-down, i.e., starting from the leaves. During this process, whenever we are trying to prove a proposition $P \Rightarrow Q$, we use the $\Rightarrow$-introduction rule backward, i.e., we add P to the set of active premises and we try to prove Q from this new set of premises. At some point, we get stuck with an atomic proposition, say R. Call the resulting deduction $\mathcal{D}_{bu}$; note that R is the only active (undischarged) premises of $\mathcal{D}_{bu}$ and the node labeled R immediately below it plays a special role; we will call it the special node of $\mathcal{D}_{bu}$. The trick is to now switch strategy and start building a proof tree top-down, starting from the leaves, using the $\Rightarrow$-elimination rule. If everything works out well, we get a deduction with root R, say $\mathcal{D}_{td}$, and then we glue this deduction $\mathcal{D}_{td}$ to the deduction $\mathcal{D}_{bu}$ in such a way that the root of $\mathcal{D}_{td}$ is identified with the special node of $\mathcal{D}_{bu}$ labeled R. We also have to make sure that all the discharged premises are linked to the correct instance of the $\Rightarrow$-introduction rule that caused them to be discharged. One of the difficulties is that during the bottom-up process, we don't know how many copies of a premise need to be discharged in a single step. We only find out how many copies of a premise need to be discharged during the top-down process.

Here is an illustration of this method for our third example. At the end of the bottom-up process, we get the deduction tree $\mathcal{D}_{bu}$:

$$\dfrac{\dfrac{\dfrac{\dfrac{(A \Rightarrow (B \Rightarrow C))^z \qquad (A \Rightarrow B)^y \qquad A^x \qquad C}{C}}{A \Rightarrow C}\; x}{(A \Rightarrow B) \Rightarrow (A \Rightarrow C)}\; y}{\big(A \Rightarrow (B \Rightarrow C)\big) \Rightarrow \big((A \Rightarrow B) \Rightarrow (A \Rightarrow C)\big)}\; z$$

At the end of the top-down process, we get the deduction tree $\mathcal{D}_{td}$:

$$\dfrac{\dfrac{A \Rightarrow (B \Rightarrow C) \qquad A}{B \Rightarrow C} \qquad \dfrac{A \Rightarrow B \qquad A}{B}}{C}$$

Finally, after gluing $\mathcal{D}_{td}$ on top of $\mathcal{D}_{bu}$ (which has the correct number of premises to be discharged), we get our proof tree:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{(A\Rightarrow(B\Rightarrow C))^z \qquad A^x}{B\Rightarrow C} \qquad \dfrac{(A\Rightarrow B)^y \qquad A^x}{B}}{C}}{A\Rightarrow C}\;x}{(A\Rightarrow B)\Rightarrow(A\Rightarrow C)}\;y}{\big(A\Rightarrow(B\Rightarrow C)\big)\Rightarrow\big((A\Rightarrow B)\Rightarrow(A\Rightarrow C)\big)}\;z$$

Let us return to the functional interpretation of implication by giving an example. The proposition $P\Rightarrow((P\Rightarrow Q)\Rightarrow Q)$ has the following proof:

$$\dfrac{\dfrac{\dfrac{(P\Rightarrow Q)^x \qquad P^y}{Q}}{(P\Rightarrow Q)\Rightarrow Q}\;x}{P\Rightarrow((P\Rightarrow Q)\Rightarrow Q)}\;y$$

Now, say P is the proposition $R\Rightarrow R$, which has the proof

$$\dfrac{\dfrac{R^z}{R}}{R\Rightarrow R}\;z$$

Using $\Rightarrow$-elimination, we obtain a proof of $((R\Rightarrow R)\Rightarrow Q)\Rightarrow Q$ from the proof of $(R\Rightarrow R)\Rightarrow(((R\Rightarrow R)\Rightarrow Q)\Rightarrow Q)$ and the proof of $R\Rightarrow R$:

$$\dfrac{\dfrac{\dfrac{\dfrac{((R\Rightarrow R)\Rightarrow Q)^x \qquad (R\Rightarrow R)^y}{Q}}{((R\Rightarrow R)\Rightarrow Q)\Rightarrow Q}\;x}{(R\Rightarrow R)\Rightarrow(((R\Rightarrow R)\Rightarrow Q)\Rightarrow Q)}\;y \qquad \dfrac{\dfrac{R^z}{R}}{R\Rightarrow R}\;z}{((R\Rightarrow R)\Rightarrow Q)\Rightarrow Q}$$

Note that the above proof is redundant. A more direct proof can be obtained as follows: Undo the last $\Rightarrow$-introduction in the proof of $(R\Rightarrow R)\Rightarrow(((R\Rightarrow R)\Rightarrow Q)\Rightarrow Q)$:

$$\cfrac{\cfrac{((R \Rightarrow R) \Rightarrow Q)^x \qquad R \Rightarrow R}{Q}}{((R \Rightarrow R) \Rightarrow Q) \Rightarrow Q}\; x$$

and then glue the proof of $R \Rightarrow R$ on top of the leaf $R \Rightarrow R$, obtaining the desired proof of $((R \Rightarrow R) \Rightarrow Q) \Rightarrow Q$:

$$\cfrac{\cfrac{((R \Rightarrow R) \Rightarrow Q)^x \qquad \cfrac{\cfrac{R^z}{R}}{R \Rightarrow R}\; z}{Q}}{((R \Rightarrow R) \Rightarrow Q) \Rightarrow Q}\; x$$

In general, one has to exercise care with the label variables. It may be necessary to rename some of these variables to avoid clashes. What we have above is an example of *proof substitution* also called *proof normalization*. We will come back to this topic in Section 1.9.

The process of discharging premises when constructing a deduction is admittedly a bit confusing. Part of the problem is that a deduction tree really represents the last of a sequence of stages (corresponding to the application of inference rules) during which the current set of "active" premises, that is, those premises that have not yet been discharged (closed, cancelled) evolves (in fact, shrinks). Some mechanism is needed to keep track of which premises are no longer active and this is what this business of labeling premises with variables achieves. Historically, this is the first mechanism that was invented. However, Gentzen (in the 1930's) came up with an alternative solution which is mathematically easier to handle. Moreover, it turns out that this notation is also better suited to computer implementations, if one wishes to implement an automated theorem prover.

The point is to keep a record of all undischarged assumptions at every stage of the deduction. Thus, a deduction is now a tree whose nodes are labeled with expressions of the form $\Gamma \rightarrow P$, called *sequents*, where P is a proposition, and Γ is a record of all undischarged assumptions at the stage of the deduction associated with this node.

During the construction of a deduction tree, it is necessary to discharge packets of assumptions consisting of one or more occurrences of the same proposition. To this effect, it is convenient to tag packets of assumptions with labels, in order to discharge the propositions in these packets in a single step. We use variables for the labels, and a packet labeled with x consisting of occurrences of the proposition P is written as $x\colon P$. Thus, in a sequent $\Gamma \rightarrow P$, the expression Γ is any finite set of the form $x_1\colon P_1, \ldots, x_m\colon P_m$, where the x_i are pairwise distinct (but the P_i need not be distinct). Given $\Gamma = x_1\colon P_1, \ldots, x_m\colon P_m$, the notation $\Gamma, x\colon P$ is only well defined when $x \neq x_i$ for all i, $1 \leq i \leq m$, in which case it denotes the set $x_1\colon P_1, \ldots, x_m\colon P_m, x\colon P$.

Using sequents, the axioms and rules of Definition 1.2.2 are now expressed as follows:

Definition 1.2.2 The axioms and inference rules of the system $\mathcal{NG}_m^{\Rightarrow}$ (*implicational logic, Gentzen-sequent style (the* $\mathcal{G}$ *in* $\mathcal{NG}$ *stands for Gentzen)*) are listed below:

$$\Gamma, x\colon P \to P$$

$$\frac{\Gamma, x\colon P \to Q}{\Gamma \to P \Rightarrow Q} \quad (\Rightarrow\text{-}\mathit{intro})$$

$$\frac{\Gamma \to P \Rightarrow Q \quad \Gamma \to P}{\Gamma \to Q} \quad (\Rightarrow\text{-}\mathit{elim})$$

In an application of the rule ($\Rightarrow$-*intro*), observe that in the lower sequent, the proposition P (labeled x) is deleted from the list of premises occurring on the left-hand side of the arrow in the upper sequent. We say that the proposition P which appears as a hypothesis of the deduction is *discharged* (or *closed*). It is important to note that the ability to label packets consisting of occurrences of the same proposition with different labels is essential, in order to be able to have control over which groups of packets of assumptions are discharged simultaneously. Equivalently, we could avoid tagging packets of assumptions with variables if we assumed that in a sequent $\Gamma \to C$, the expression Γ, also called a *context*, is a *multiset* of propositions.

Below we show a proof of the third example given above in our new system. Let

$$\Gamma = x\colon A \Rightarrow (B \Rightarrow C), y\colon A \Rightarrow B, z\colon A.$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\Gamma \to A \Rightarrow (B \Rightarrow C) \qquad \Gamma \to A}{\Gamma \to B \Rightarrow C} \qquad \dfrac{\Gamma \to A \Rightarrow B \qquad \Gamma \to A}{\Gamma \to B}}{x\colon A \Rightarrow (B \Rightarrow C), y\colon A \Rightarrow B, z\colon A \to C}}{x\colon A \Rightarrow (B \Rightarrow C), y\colon A \Rightarrow B \to A \Rightarrow C}}{x\colon A \Rightarrow (B \Rightarrow C) \to (A \Rightarrow B) \Rightarrow (A \Rightarrow C)}}{\to \big(A \Rightarrow (B \Rightarrow C)\big) \Rightarrow \big((A \Rightarrow B) \Rightarrow (A \Rightarrow C)\big)}$$

Remark: An attentive reader will have surely noticed that the second version of the $\Rightarrow$-elimination rule,

$$\frac{\Gamma \to P \Rightarrow Q \quad \Gamma \to P}{\Gamma \to Q} \quad (\Rightarrow\text{-}\mathit{elim}),$$

differs slightly from the first version given in Definition 1.2.1. Indeed, in Prawitz's style, the rule that matches exactly the $\Rightarrow$-elim rule above is

$$\dfrac{\dfrac{\Gamma}{P \Rightarrow Q} \qquad \dfrac{\Gamma}{P}}{Q}$$

where the deductions of $P \Rightarrow Q$ and P have the *same* set of premises, Γ. Equivalently, the rule in sequent-format that corresponds to the $\Rightarrow$-elimination rule of Definition 1.2.1 is

$$\frac{\Gamma \to P \Rightarrow Q \quad \Delta \to P}{\Gamma, \Delta \to Q} \quad (\Rightarrow\textit{-elim'}),$$

where Γ, Δ must be interpreted as the union of Γ and Δ.

A moment of reflexion will reveal that the resulting proofs systems are equivalent (that is, every proof in one system can converted to a proof in the other system). The version of the $\Rightarrow$-elimination rule in Definition 1.2.1 may be considered preferable because it gives us the ability to make the sets of premises labeling leaves smaller. On the other hand, after experimenting with the construction of proofs, one gets the feeling that every proof can be simplified to a "unique minimal" proof, if we define "minimal" in a suitable sense, namely, that a minimal proof never contains an elimination rule immediately following an introduction rule (for more on this, see Section 1.9). Then, it turns out that to define the notion of uniqueness of proofs, the second version is preferable. However, it is important to realize that in general, a proposition may possess distinct minimal proofs!

In principle, it does not matter which of the two systems $\mathcal{N}_m^{\Rightarrow}$ or $\mathcal{NG}_m^{\Rightarrow}$ we use to construct deductions; it is basically a matter of taste. My experience is that I make fewer mistakes with the Gentzen-sequent style system $\mathcal{NG}_m^{\Rightarrow}$.

We now describe the inference rules dealing with the connectives $\wedge$, $\vee$ and $\perp$.

1.3 Adding $\wedge$, $\vee$, $\perp$; The Proof Systems $\mathcal{N}_c^{\Rightarrow,\wedge,\vee,\perp}$ and $\mathcal{NG}_c^{\Rightarrow,\wedge,\vee,\perp}$

Recall that $\neg P$ is an abbreviation for $P \Rightarrow \perp$.

Definition 1.3.1 The axioms and inference rules for *(propositional) classical logic* are:

Axioms:

$$\frac{\Gamma, P}{P}$$

The $\Rightarrow$*-introduction rule*:

$$\dfrac{\dfrac{\Gamma, P^x}{Q}}{P \Rightarrow Q}\ x$$

The $\Rightarrow$-*elimination rule*:

$$\dfrac{\dfrac{\Gamma}{P \Rightarrow Q} \qquad \dfrac{\Delta}{P}}{Q}$$

The $\wedge$-*introduction rule*:

$$\dfrac{\dfrac{\Gamma}{P} \qquad \dfrac{\Delta}{Q}}{P \wedge Q}$$

The $\wedge$-*elimination rule*:

$$\dfrac{\dfrac{\Gamma}{P \wedge Q}}{P} \qquad\qquad \dfrac{\dfrac{\Gamma}{P \wedge Q}}{Q}$$

The $\vee$-*introduction rule*:

$$\dfrac{\dfrac{\Gamma}{P}}{P \vee Q} \qquad\qquad \dfrac{\dfrac{\Gamma}{Q}}{P \vee Q}$$

The $\vee$-*elimination rule*:

$$\dfrac{\dfrac{\Gamma}{P \vee Q} \qquad \dfrac{\Delta, P^x}{R} \qquad \dfrac{\Lambda, Q^y}{R}}{R}\; {\scriptstyle x,y}$$

The $\perp$-*elimination rule*:

$$\dfrac{\dfrac{\Gamma}{\perp}}{P}$$

The *proof-by-contradiction rule* (also known as *reductio ad absurdum rule*, for short *RAA*):

$$\dfrac{\dfrac{\Gamma, \neg P^x}{\perp}}{P}\; {\scriptstyle x}$$

Since $\neg P$ is an abbreviation for $P \Rightarrow \perp$, the $\neg$-introduction rule is a special case of the $\Rightarrow$-introduction rule (with $Q = \perp$). However, it is worth stating it explicitly:

The ¬-*introduction rule*:

$$\cfrac{\cfrac{\Gamma, P^x}{\bot}}{\neg P}\ x$$

Similarly, the ¬-elimination rule is a special case of ⇒-elimination applied to $\neg P\ (= P \Rightarrow \bot)$ and P:

The ¬-*elimination rule*:

$$\frac{\cfrac{\Gamma}{\neg P} \qquad \cfrac{\Delta}{P}}{\bot}$$

In the above axioms and rules, Γ, Δ or Λ may be empty, P, Q, R denote arbitrary propositions built up from the atoms in **PS** and all the premises labeled x are discharged. A *deduction tree* is a tree whose interior nodes correspond to applications of the above inference rules. A *proof tree* is a deduction tree such that *all its premises* are discharged. The above proof system is denoted $\mathcal{N}_c^{\Rightarrow,\wedge,\vee,\bot}$ (here, the subscript c stands for *classical*).

The system obtained by removing the proof-by-contradiction (RAA) rule is called *(propositional) intuitionistic logic* and is denoted $\mathcal{N}_i^{\Rightarrow,\wedge,\vee,\bot}$. The system obtained by deleting both the ⊥-elimination rule and the proof-by-contradiction rule is called *(propositional) minimal logic* and is denoted $\mathcal{N}_m^{\Rightarrow,\wedge,\vee,\bot}$.

The version of $\mathcal{N}_c^{\Rightarrow,\wedge,\vee,\bot}$ in terms of Gentzen sequents is the following:

Definition 1.3.2 The axioms and inference rules of the system $\mathcal{NG}_i^{\Rightarrow,\wedge,\vee,\bot}$ (of *propositional classical logic, Gentzen-sequent style*) are listed below:

$$\Gamma, x\colon P \to P$$

$$\frac{\Gamma, x\colon P \to Q}{\Gamma \to P \Rightarrow Q}\quad (\Rightarrow\text{-}intro)$$

$$\frac{\Gamma \to P \Rightarrow Q \quad \Gamma \to P}{\Gamma \to Q}\quad (\Rightarrow\text{-}elim)$$

$$\frac{\Gamma \to P \quad \Gamma \to Q}{\Gamma \to P \wedge Q}\quad (\wedge\text{-}intro)$$

$$\frac{\Gamma \to P \wedge Q}{\Gamma \to P}\quad (\wedge\text{-}elim) \qquad \frac{\Gamma \to P \wedge Q}{\Gamma \to Q}\quad (\wedge\text{-}elim)$$

$$\frac{\Gamma \to P}{\Gamma \to P \vee Q}\quad (\vee\text{-}intro) \qquad \frac{\Gamma \to Q}{\Gamma \to P \vee Q}\quad (\vee\text{-}intro)$$

$$\frac{\Gamma \to P \vee Q \quad \Gamma, x\colon P \to R \quad \Gamma, y\colon Q \to R}{\Gamma \to R} \quad (\vee\text{-}elim)$$

$$\frac{\Gamma \to \perp}{\Gamma \to P} \quad (\perp\text{-}elim)$$

$$\frac{\Gamma, x\colon \neg P \to \perp}{\Gamma \to P} \quad (by\text{-}contra)$$

$$\frac{\Gamma, x\colon P \to \perp}{\Gamma \to \neg P} \quad (\neg\text{-introduction})$$

$$\frac{\Gamma \to \neg P \quad \Gamma \to P}{\Gamma \to \perp} \quad (\neg\text{-elimination})$$

Since the rule ($\perp$-*elim*) is trivial (does nothing) when $P = \perp$, from now on, we will assume that $P \neq \perp$. *Propositional minimal logic*, denoted $\mathcal{NG}_m^{\Rightarrow,\wedge,\vee,\perp}$, is obtained by dropping the ($\perp$-*elim*) and (*by-contra*) rules. *Propositional intuitionistic logic*, denoted $\mathcal{NG}_i^{\Rightarrow,\wedge,\vee,\perp}$, is obtained by dropping the (*by-contra*) rule.

When we say that a proposition, P, is *provable from* Γ, we mean that we can construct a proof tree whose conclusion is P and whose set of premises is Γ, in one of the systems $\mathcal{N}_c^{\Rightarrow,\wedge,\vee,\perp}$ or $\mathcal{NG}_c^{\Rightarrow,\wedge,\vee,\perp}$. Therefore, when we use the word "provable" unqualified, we mean provable in *classical logic*. If P is provable from Γ in one of the intuitionistic systems $\mathcal{N}_i^{\Rightarrow,\wedge,\vee,\perp}$ or $\mathcal{NG}_i^{\Rightarrow,\wedge,\vee,\perp}$, then we say *intuitionistically provable* (and similarly, if P is provable from Γ in one of the systems $\mathcal{N}_m^{\Rightarrow,\wedge,\vee,\perp}$ or $\mathcal{NG}_m^{\Rightarrow,\wedge,\vee,\perp}$, then we say *provable in minimal logic*). When P is provable from Γ, most people write $\Gamma \vdash P$, or $\vdash \Gamma \to P$, sometimes with the name of the corresponding proof system tagged as a subscript on the sign $\vdash$ if necessary to avoid ambiguities. When Γ is empty, we just say P is provable (provable in intuitionistic logic, *etc.*) and write $\vdash P$.

We treat *logical equivalence* as a derived connective, that is, we view $P \equiv Q$ as an abbreviation for $(P \Rightarrow Q) \wedge (Q \Rightarrow P)$. In view of the inference rules for $\wedge$, we see that to prove a logical equivalence $P \equiv Q$, we just have to prove both implications $P \Rightarrow Q$ and $Q \Rightarrow P$.

In view of the $\neg$-elimination rule, we may be tempted to interpret the provability of a negation, $\neg P$, is as "P is not provable". Indeed, if $\neg P$ and P were both provable, then $\perp$ would be provable. So, P should not be provable if $\neg P$ is. However, if P is not provable, then $\neg P$ is **not** provable in general! There are plenty of propositions such that neither P nor $\neg P$ is provable (for instance, P, with P an atomic proposition). Thus, the fact that P is not provable is not equivalent to the provability of $\neg P$ and we should not interpret $\neg P$ as "P is not provable".

Let us now make some (much-needed) comments about the above inference rules. There is no need to repeat our comments regarding the $\Rightarrow$-rules.

The $\wedge$-introduction rule says that in order to prove a conjunction $P \wedge Q$ from some premises Γ, all we have to do is to prove *both* that P is provable from Γ *and* that Q is provable from Γ. The $\wedge$-elimination rule says that once we have proved $P \wedge Q$ from Γ, then P (and Q) is also provable from Γ. This makes sense intuitively as $P \wedge Q$ is "stronger" than P and Q separately ($P \wedge Q$ is true iff both P and Q are true).

The $\vee$-introduction rule says that if P (or Q) has been proved from Γ, then $P \vee Q$ is also provable from Γ. Again, this makes sense intuitively as $P \vee Q$ is "weaker" than P and Q. The $\vee$-elimination rule formalizes the *proof-by-cases* method. It is a more subtle rule. The idea is that if we know that in the case where P is already assumed to be provable and similarly in the case where Q is already assumed to be provable that we can prove R (also using premises in Γ), then if $P \vee Q$ is also provable from Γ, as we have "covered both cases", it should be possible to prove R from Γ only (i.e., the premises P and Q are discarded).

The $\perp$-elimination rule formalizes the principle that once a false statement has been established, then anything should be provable.

The proof-by-contradiction rule formalizes the method of proof by contradiction! That is, in order to prove that P can be deduced from some premises Γ, one may assume the negation, $\neg P$, of P (intuitively, assume that P is false) and then derive a contradiction from Γ and $\neg P$ (i.e., derive falsity). Then, P actually follows from Γ *without using* $\neg P$ *as a premise*, i.e., $\neg P$ is discharged.

Most people, I believe, will be comfortable with the rules of minimal logic and will agree that they constitute a "reasonable" formalization of the rules of reasoning involving $\Rightarrow$, $\wedge$ and $\vee$. Indeed, these rules seem to express the intuitive meaning of the connectives $\Rightarrow$, $\wedge$ and $\vee$. However, some may question the two rules $\perp$-elimination and proof-by-contradiction. Indeed, their meaning is not as clear and, certainly, the proof-by-contradiction rule introduces a form of indirect reasoning that is somewhat worrisome.

The problem has to do with the meaning of disjunction and negation and more generally, with the notion of *constructivity* in mathematics. In fact, in the early 1900's, some mathematicians, especially L. Brouwer (1881-1966), questioned the validity of the proof-by-contradiction rule, among other principles. Two specific cases illustrate the problem, namely, the propositions

$$P \vee \neg P \quad \text{and} \quad \neg\neg P \Rightarrow P.$$

As we will see shortly, the above propositions are both provable in classical logic. Now, Brouwer and some mathematicians belonging to his school of thoughts (the so-called "intuitionsists" or "constructivists") advocate that in order to prove a disjunction, $P \vee Q$ (from some premises Γ) one has to either exhibit a proof of P or a proof or Q (from Γ). However, it can be shown that this fails for $P \vee \neg P$. The fact that $P \vee \neg P$ is provable (in classical logic) **does not** imply (in general) that either P is provable or that $\neg P$ is provable! That $P \vee \neg P$ is provable is sometimes called the *principle of the excluded middle*! In intuitionistic logic,

$P \vee \neg P$ is **not** provable (in general). Of course, if one gives up the proof-by-contradiction rule, then fewer propositions become provable. On the other hand, one may claim that the propositions that remain provable have more constructive proofs and thus, feels on safer grounds.

A similar controversy arises with $\neg\neg P \Rightarrow P$. If we give up the proof-by-contradiction rule, then this formula is no longer provable, i.e., $\neg\neg P$ is no longer equivalent to P. Perhaps this relates to the fact that if one says

" I don't have no money"

then this does not mean that this person has money! (Similarly with "I don't get no satisfaction", ...). However, note that one can still prove $P \Rightarrow \neg\neg P$ in minimal logic (try doing it!). Even stranger, $\neg\neg\neg P \Rightarrow \neg P$ is provable in intuitionistic (and minimal) logic, so $\neg\neg\neg P$ and $\neg P$ are equivalent intuitionistically!

Remark: Suppose we have a deduction

$$\frac{\Gamma, \neg P}{\perp}$$

as in the proof by contradiction rule. Then, by $\neg$-introduction, we get a deduction of $\neg\neg P$ from Γ:

$$\cfrac{\cfrac{\Gamma, \neg P^x}{\perp}}{\neg\neg P}\; x$$

So, if we knew that $\neg\neg P$ was equivalent to P (actually, if we knew that $\neg\neg P \Rightarrow P$ is provable) then the proof by contradiction rule would be justified as a valid rule (it follows from modus ponens). We can view the proof by contradiction rule as a sort of act of faith that consists in saying that if we can derive an inconsistency (i.e., chaos) by assuming the falsity of a statement P, then P has to hold in the first place. It not so clear that such an act of faith is justified and the intuitionists refuse to take it!

Constructivity in mathematics is a fascinating subject but it is a topic that is really outside the scope of this course. What we hope is that our brief and very incomplete discussion of constructivity issues made the reader aware that the rules of logic are not cast in stone and that, in particular, there isn't **only one** logic.

We feel safe in saying that most mathematicians work with classical logic and only few of them have reservations about using the proof-by-contradiction rule. Nevertherless, intuitionistic logic has its advantages, especially when it comes to proving the correctess of programs (a branch of computer science!). We will come back to this point several times in this course.

In the rest of this section, we make further useful remarks about (classical) logic and give some explicit examples of proofs illustrating the inference rules of classical logic. We begin by proving that $P \vee \neg P$ is provable in classical logic.

Proposition 1.3.3 *The proposition $P \vee \neg P$ is provable in classical logic.*

Proof. We prove that $P \vee (P \Rightarrow \perp)$ is provable by using the proof-by-contradiction rule as shown below:

$$\dfrac{\dfrac{((P \vee (P \Rightarrow \perp)) \Rightarrow \perp)^y \qquad \dfrac{\dfrac{\dfrac{((P \vee (P \Rightarrow \perp)) \Rightarrow \perp)^y \qquad \dfrac{P^x}{P \vee (P \Rightarrow \perp)}}{\perp}}{P \Rightarrow \perp}\; x}{P \vee (P \Rightarrow \perp)}}{\perp}}{P \vee (P \Rightarrow \perp)}\; y \text{ (by-contra)}$$

□

Next, we consider the equivalence of P and $\neg\neg P$.

Proposition 1.3.4 *The proposition $P \Rightarrow \neg\neg P$ is provable in minimal logic. The proposition $\neg\neg P \Rightarrow P$ is provable in classical logic. Therefore, in classical logic, P is equivalent to $\neg\neg P$.*

Proof. We leave that $P \Rightarrow \neg\neg P$ is provable in minimal logic as an exercise. Below is a proof of $\neg\neg P \Rightarrow P$ using the proof-by-contradiction rule:

$$\dfrac{\dfrac{\dfrac{((P \Rightarrow \perp) \Rightarrow \perp)^y \qquad (P \Rightarrow \perp)^x}{\perp}}{P}\; x \text{ (by-contra)}}{((P \Rightarrow \perp) \Rightarrow \perp) \Rightarrow P}\; y$$

□

The next proposition shows why $\perp$ can be viewed as the "ultimate" contradiction.

Proposition 1.3.5 *In intuitionistic logic, the propositions $\perp$ and $P \wedge \neg P$ are equivalent for all P. Thus, $\perp$ and $P \wedge \neg P$ are also equivalent in classical propositional logic*

Proof. We need to show that both $\perp \Rightarrow (P \wedge \neg P)$ and $(P \wedge \neg P) \Rightarrow \perp$ are provable in intuitionistic logic. The provability of $\perp \Rightarrow (P \wedge \neg P)$ is an immediate consequence or $\perp$-elimination, with $\Gamma = \emptyset$. For $(P \wedge \neg P) \Rightarrow \perp$, we have the following proof:

$$\dfrac{\dfrac{\dfrac{(P \land \neg P)^x}{\neg P} \qquad \dfrac{(P \land \neg P)^x}{P}}{\bot}}{(P \land \neg P) \Rightarrow \bot}\; x$$

□

So, in intuitionistic logic (and also in classical logic), $\bot$ is equivalent to $P \land \neg P$ for all P. This means that $\bot$ is the "ultimate" contradiction, it corresponds to total inconsistency. By the way, we could have the bad luck that the system $\mathcal{N}_c^{\Rightarrow,\land,\lor,\bot}$ (or $\mathcal{N}_i^{\Rightarrow,\land,\lor,\bot}$ or even $\mathcal{N}_m^{\Rightarrow,\land,\lor,\bot}$) is *inconsistent*, that is, that $\bot$ is provable! Fortunately, this is not the case, although this is hard to prove. (It is also the case that $P \lor \neg P$ and $\neg\neg P \Rightarrow P$ are **not** provable in intuitionistic logic, but this too is hard to prove!)

1.4 Clearing Up Differences Between ¬-introduction, ⊥-elimination and RAA

The differences between the rules, ¬-introduction, ⊥-elimination and the proof by contradiction rule (RAA) are often unclear to the uninitiated reader and this tends to cause confusion. In this section, we will try to clear up some common misconceptions about these rules.

Confusion 1. Why is RAA not a special case of ¬-introduction?

$$\dfrac{\dfrac{\Gamma, P^x}{\bot}}{\neg P}\; x\,(\neg\text{-intro}) \qquad\qquad \dfrac{\dfrac{\Gamma, \neg P^x}{\bot}}{P}\; x\,(\text{RAA})$$

The only apparent difference between ¬-introduction (on the left) and RAA (on the right) is that in RAA, the premise P is negated but the conclusion is not, whereas in ¬-introduction the premise P is not negated but the conclusion is.

The important difference is that the conclusion of RAA is **not** negated. If we had applied ¬-introduction instead of RAA on the right, we would have obtained

$$\dfrac{\dfrac{\Gamma, \neg P^x}{\bot}}{\neg\neg P}\; x\,(\neg\text{-intro})$$

where the conclusion would have been $\neg\neg P$ as opposed to P. However, as we already said earlier, $\neg\neg P \Rightarrow P$ is **not** provable intuitionistically. Consequenly, RAA **is not** a special case of ¬-introduction.

Confusion 2. Is there any difference between ⊥-elimination and RAA?

$$\frac{\dfrac{\Gamma}{\perp}}{P}\ (\perp\text{-elim}) \qquad\qquad \frac{\dfrac{\Gamma, \neg P^x}{\perp}}{P}\ {\scriptstyle x}\,(\text{RAA})$$

The difference is that $\perp$-elimination does not discharge any of its premises. In fact, RAA is a stronger rule which implies $\perp$-elimination as we now demonstate.

RAA implies $\perp$-elimination.

Suppose we have a deduction

$$\frac{\Gamma}{\perp}$$

Then, for any proposition P, we can add the premise $\neg P$ to every leaf of the above deduction tree and we get the deduction tree

$$\frac{\Gamma, \neg P}{\perp}$$

We can now apply RAA to get the following deduction tree of P from Γ (since $\neg P$ is discharged), and this is just the result of $\perp$-elimination:

$$\frac{\dfrac{\Gamma, \neg P^x}{\perp}}{P}\ {\scriptstyle x}\,(\text{RAA})$$

The above considerations also show that RAA is obtained from $\neg$-introduction by adding the new rule of $\neg\neg$-*elimination*:

$$\frac{\dfrac{\Gamma}{\neg\neg P}}{P}\ (\neg\neg\text{-elimination})$$

Some authors prefer adding the $\neg\neg$-elimination rule to intuitionistic logic instead of RAA in order to obtain classical logic. As we just demonstrated, the two additions are equivalent: by adding either RAA or $\neg\neg$-elimination to intuitionistic logic, we get classical logic.

There is another way to obtain RAA from the rules of intuitionistic logic, this time, using the propositions of the form $P \vee \neg P$. We saw in Proposition 1.3.3 that all formulae of the form $P \vee \neg P$ are provable in classical logic (using RAA).

Confusion 3. Are propositions of the form $P \vee \neg P$ provable in intuitionistic logic?

The answer is **no**, which may be disturbing to some readers. In fact, it is quite difficult to prove that propositions of the form $P \vee \neg P$ are not provable in intuitionistic logic. One method consists in using the fact that intuitionistic proofs can be normalized (see Section 1.9 for more on normalization of proofs). Another method uses Kripke models (see Section 1.7 and van Dalen [44]).

Part of the difficulty in understanding at some intuitive level why propositions of the form $P \vee \neg P$ are not provable in intuitionistic logic is that the notion of truth based on the truth values **true** and **false** is deeply rooted in all of us. In this frame of mind, it seems ridiculous to question the provability of $P \vee \neg P$, since its truth value is **true** whether P is assigned the value **true** or **false**. Classical two-valued truth values semantics is too crude for intuitionistic logic.

Another difficulty is that it is tempting to equate the notion of truth and the notion of provability. Unfortunately, because classical truth values semantics is too crude for intuitionistic logic, there are propositions that are universally true (i.e., they evaluate to **true** for all possible truth assignments of the atomic letters in them) and yet they are **not** provable intuitionistically. The propositions $P \vee \neg P$ and $\neg\neg P \Rightarrow P$ are such examples.

One of the major motivations for advocating intuitionistic logic is that it yields proofs that are more constructive than classical proofs. For example, in classical logic, when we prove a disjunction $P \vee Q$, we generally can't conclude that either P or Q is provable, as examplified by $P \vee \neg P$. A more interesting example involving a non-constructive proof of a disjunction will be given in Section 1.5. But, in intuitionistic logic, from a proof of $P \vee Q$, it is possible to extract either a proof of P or a proof or Q (and similarly for existential statements, see Section 1.8). This property is not easy to prove. It is a consequence of the normal form for intuitionistic proofs (see Section 1.9).

In brief, besides being a fun intellectual game, intuitionistic logic is only an interesting alternative to classical logic if we care about the constructive nature of our proofs. But then, we are forced to abandon the classical two-valued truth values semantics and adopt other semantics such as Kripke semantics. If we do not care about the constructive nature of our proofs and if we want to stick to two-valued truth values semantics, then we should stick to classical logic. Most people do that, so don't feel bad if you are not comfortable with intuitionistic logic!

One way to gauge how intuitionisic logic differs from classical logic is to ask what kind of propositions need to be added to intuitionisic logic in order to get classical logic. It turns out that if all the propositions of the form $P \vee \neg P$ are considered to be axioms, then RAA follows from some of the rules of intuitionistic logic.

RAA holds in Intuitionistic logic + all axioms $P \vee \neg P$.

The proof involves a subtle use of the $\perp$-elimination and $\vee$-elimination rules which may be a bit puzzling. Assume, as we do when when use the proof by contradiction rule (RAA) that we have a deduction

$$\frac{\Gamma, \neg P}{\perp}$$

Here is the deduction tree demonstrating that RAA is a derived rule:

$$\dfrac{P \lor \neg P \qquad \dfrac{P^x}{P} \qquad \dfrac{\dfrac{\Gamma, \neg P^y}{\bot}}{P}\ (\bot\text{-elim})}{P}\ {}_{x,y}\ (\lor\text{-elim})$$

At first glance, the rightmost subtree

$$\dfrac{\dfrac{\Gamma, \neg P^y}{\bot}}{P}\ (\bot\text{-elim})$$

appears to use RAA and our argument looks circular! But this is not so because the premise $\neg P$ labeled y is *not* discharged in the step that yields P as conclusion; the step that yields P is a $\bot$-elimination step. The premise $\neg P$ labeled y is actually discharged by the $\lor$-elimination rule (and so is the premise P labeled x). So, our argument establishing RAA is not circular after all!

In conclusion, intuitionistic logic is obtained from classical logic by *taking away the proof by contradiction rule (RAA)*. In this more restrictive proof system, we obtain more constructive proofs. In that sense, the situation is better than in classical logic. The major drawback is that we can't think in terms of classical truth values semantics anymore.

Conversely, classical logic is obtained from intuitionistic logic in at least three ways:

1. Add the proof by contradiction rule (RAA).
2. Add the $\neg\neg$-elimination rule.
3. Add all propositions of the form $P \lor \neg P$ as axioms.

1.5 Other Rules of Classical Logic and Examples of Proofs

In classical logic, we have the de Morgan laws:

Proposition 1.5.1 *The following equivalences (de Morgan laws) are provable in classical logic:*

$$\neg(P \land Q) \equiv \neg P \lor \neg Q$$
$$\neg(P \lor Q) \equiv \neg P \land \neg Q.$$

In fact, $\neg(P \vee Q) \equiv \neg P \wedge \neg Q$ *and* $(\neg P \vee \neg Q) \Rightarrow \neg(P \wedge Q)$ *are provable in intuitionistic logic. The proposition* $(P \wedge \neg Q) \Rightarrow \neg(P \Rightarrow Q)$ *is provable in intuitionistic logic and* $\neg(P \Rightarrow Q) \Rightarrow (P \wedge \neg Q)$ *is provable in classical logic. Therefore,* $\neg(P \Rightarrow Q)$ *and* $P \wedge \neg Q$ *are equivalent in classical logic. Furthermore,* $P \Rightarrow Q$ *and* $\neg P \vee Q$ *are equivalent in classical logic and* $(\neg P \vee Q) \Rightarrow (P \Rightarrow Q)$ *is provable in intuitionistic logic.*

Proof. Here is an intuitionistic proof of $(\neg P \vee Q) \Rightarrow (P \Rightarrow Q)$:

$$\dfrac{\dfrac{(\neg P \vee Q)^w \qquad \dfrac{\dfrac{\dfrac{\neg P^z \qquad P^x}{\bot}}{Q}}{P \Rightarrow Q}\;{\scriptstyle x} \qquad \dfrac{\dfrac{P^y \qquad Q^t}{Q}}{P \Rightarrow Q}\;{\scriptstyle y}}{P \Rightarrow Q}\;{\scriptstyle z,t}}{(\neg P \vee Q) \Rightarrow (P \Rightarrow Q)}\;{\scriptstyle w}$$

Here is a classical proof of $(P \Rightarrow Q) \Rightarrow (\neg P \vee Q)$:

$$\dfrac{\dfrac{\dfrac{(\neg(\neg P \vee Q))^y \qquad \dfrac{\dfrac{(P \Rightarrow Q)^z \qquad \dfrac{\dfrac{(\neg(\neg P \vee Q))^y \qquad \dfrac{\neg P^x}{\neg P \vee Q}}{\bot}}{P}\;{\scriptstyle x}\ \mathrm{RAA}}{Q}}{\neg P \vee Q}}{\bot}}{\neg P \vee Q}\;{\scriptstyle y}\ \mathrm{RAA}}{(P \Rightarrow Q) \Rightarrow (\neg P \vee Q)}\;{\scriptstyle z}$$

The other proofs are left as exercises. □

Propositions 1.3.4 and 1.5.1 show a property that is very specific to classical logic, namely, that the logical connectives $\Rightarrow, \wedge, \vee, \neg$ are not independent. For example, we have $P \wedge Q \equiv \neg(\neg P \vee \neg Q)$, which shows that $\wedge$ can be expressed in terms of $\vee$ and $\neg$. In intuitionistic logic, $\wedge$ and $\vee$ cannot be expressed in terms of each other via negation.

The fact that the logical connectives $\Rightarrow, \wedge, \vee, \neg$ are not independent in classical logic suggests the following question: Are there propositions, written in terms of $\Rightarrow$ only, that are provable classically but not provable intuitionistically?

The answer is yes! For instance, the proposition $((P \Rightarrow Q) \Rightarrow P) \Rightarrow P$ (known as *Peirce's law*) is provable classically (do it) but it can be shown that it is not provable intuitionistically.

In addition to the proof by cases method and the proof by contradiction method, we also have the proof by contrapositive method valid in classical logic:

Proof by contrapositive rule:

$$\dfrac{\dfrac{\Gamma, \neg Q^x}{\neg P}}{P \Rightarrow Q}\; x$$

This rule says that in order to prove an implication $P \Rightarrow Q$ (from Γ), one may assume $\neg Q$ as proved, and then deduce that $\neg P$ is provable from Γ and $\neg Q$. This inference rule is valid in classical logic because we can construct the following proof:

$$\dfrac{\dfrac{\dfrac{\dfrac{\Gamma, \neg Q^x}{\neg P} \qquad P^y}{\bot}}{Q}\; x \text{ (by-contra)}}{P \Rightarrow Q}\; y$$

We will now give some explicit examples of proofs illustrating the proof principles that we just discussed.

Recall that the *set of integers* is the set

$$\mathbb{Z} = \{\cdots, -2, -1, 0, 1, 2, \cdots\}$$

and that the *set of natural numbers* is the set

$$\mathbb{N} = \{0, 1, 2, \cdots\}.$$

(Some authors exclude 0 from $\mathbb{N}$. We don't like this discrimination against zero.) An integer is *even* if it is divisible by 2, that is, if it can be written as $2k$, where $k \in \mathbb{Z}$. An integer is *odd* if it is not divisible by 2, that is, if it can be written as $2k + 1$, where $k \in \mathbb{Z}$. The following facts are essentially obvious:

(a) The sum of even integers is even.

(b) The sum of an even integer and of an odd integer is odd.

(c) The sum of two odd integers is even.

(d) The product of odd integers is odd.

(e) The product of an even integer with any integer is even.

Now, we prove the following fact using the proof by cases method.

Proposition 1.5.2 *Let a, b, c be odd integers. For any integers p and q, if p and q are not both even, then*

$$ap^2 + bpq + cq^2$$

is odd.

Proof. We consider the three cases:

1. p and q are odd. In this case as a, b and c are odd, by (d) all the products ap^2, bpq and cq^2 are odd. By (c), $ap^2 + bpq$ is even and by (b), $ap^2 + bpq + cq^2$ is odd.

2. p is even and q is odd. In this case, by (e), both ap^2 and bpq are even and by (d), cq^2 is odd. But then, by (a), $ap^2 + bpq$ is even and by (b), $ap^2 + bpq + cq^2$ is odd.

3. p is odd and q is even. This case is analogous to the previous case, except that p and q are interchanged. The reader should have no trouble filling in the details.

Since all three cases exhaust all possibilities for p and q not to be both even, the proof is complete by the $\vee$-elimination rule (applied twice). □

The set of rational numbers $\mathbb{Q}$ consists of all fractions p/q, where $p, q \in \mathbb{Z}$, with $q \neq 0$. We now use Proposition 1.5.2 and the proof by contradiction method to prove

Proposition 1.5.3 *Let a, b, c be odd integers. Then, the equation*

$$aX^2 + bX + c = 0$$

has no rational solution X.

Proof. We proceed by contradiction (by this, we mean that we use the proof by contradiction rule). So, assume that there is a rational solution $X = p/q$. We may assume that p and q have no common divisor, which implies that p and q are not both even. As $q \neq 0$, if $aX^2 + bX + c = 0$, then by multiplying by q^2, we get

$$ap^2 + bpq + cq^2 = 0.$$

However, as p and q are not both even and a, b, c are odd, we know from Proposition 1.5.2 that $ap^2 + bpq + cq^2$ is odd. This contradicts the fact that $p^2 + bpq + cq^2 = 0$ and thus, finishes the proof. □

As as example of the proof by contrapositive method, we prove that if an integer n^2 is even, then n must be even.

Observe that if an integer is not even then it is odd (and vice-versa). Thus, the contrapositive of our statement is: If n is odd, then n^2 is odd. But, to say that n is odd is to say that $n = 2k + 1$ and then, $n^2 = (2k + 1)^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1$, which shows that n^2 is odd.

A real number $a \in \mathbb{R}$ is said to be *irrational* if it cannot be expressed as a number in $\mathbb{Q}$ (a fraction). The reader should prove that $\sqrt{2}$ is irrational by adapting the arguments used in the two previous propositions.

Remark: Let us return briefly to the issue of constructivity in classical logic, in particular when it comes to disjunctions. Consider the question: are there two irrational real numbers a and b such that a^b is rational? Here is a way to prove that this indeed the case. Consider the number $\sqrt{2}^{\sqrt{2}}$. If this number is rational, then $a = \sqrt{2}$ and $b = \sqrt{2}$ is an answer to our question (since we already know that $\sqrt{2}$ is irrational). Now, observe that

$$(\sqrt{2}^{\sqrt{2}})^{\sqrt{2}} = \sqrt{2}^{\sqrt{2}\times\sqrt{2}} = \sqrt{2}^2 = 2 \quad \text{is rational!}$$

Thus, if $\sqrt{2}^{\sqrt{2}}$ is irrational, then $a = \sqrt{2}^{\sqrt{2}}$ and $b = \sqrt{2}$ is an answer to our question. So, we proved that

($\sqrt{2}$ is irrational and $\sqrt{2}^{\sqrt{2}}$ is rational) or

($\sqrt{2}^{\sqrt{2}}$ and $\sqrt{2}$ are irrational and $(\sqrt{2}^{\sqrt{2}})^{\sqrt{2}}$ is rational).

However, the above proof does not tell us whether $\sqrt{2}^{\sqrt{2}}$ is rational or not!

We see one of the shortcomings of classical reasoning: certain statements (in particular, disjunctive or existential) are provable but their proof does provide an explicit answer. It is in that sense that classical logic is not constructive.

Many more examples of non-constructive arguments in classical logic can be given.

1.6 Truth Values Semantics for Classical Logic Soundness and Completeness

So far, even though we have deliberately focused on proof theoy and ignored semantic issues, we feel that we can't postpone any longer a discussion of the truth values semantics for classical propositional logic.

We all learned early on that the logical connectives, $\Rightarrow$, $\wedge$, $\vee$ and $\neg$ can be interpreted as boolean functions, that is, functions whose arguments and whose values range over the set of *truth values*,

$$\mathbf{BOOL} = \{\mathbf{true}, \mathbf{false}\}.$$

These functions are given by the following *truth tables*:

P	Q	$P \Rightarrow Q$	$P \wedge Q$	$P \vee Q$	$\neg P$
true	**true**	**true**	**true**	**true**	**false**
true	**false**	**false**	**false**	**true**	**false**
false	**true**	**true**	**false**	**true**	**true**
false	**false**	**true**	**false**	**false**	**true**

Now, any proposition, P, built up over the set of atomic propositions, **PS**, (our propositional symbols) contains a finite set of propositional letters, say

$$\{P_1, \ldots, P_m\}.$$

If we assign some truth value (from **BOOL**) to each symbol, P_i, then we can "compute" the *truth value* of P under this assignment by using (recursively) the truth tables above. For example, the proposition $\mathbf{P}_1 \Rightarrow (\mathbf{P}_1 \Rightarrow \mathbf{P}_2)$, under the truth assignment,

$$\mathbf{P}_1 = \mathbf{true},\ \mathbf{P}_2 = \mathbf{false},$$

evaluates to **false**. However, under the truth assignment,

$$\mathbf{P}_1 = \mathbf{true},\ \mathbf{P}_2 = \mathbf{true},$$

our proposition evaluates to **true**.

If we now consider the proposition,

$$P = (\mathbf{P}_1 \Rightarrow (\mathbf{P}_2 \Rightarrow \mathbf{P}_1)),$$

then it is easy to see that P evaluates to **true** for all four possible truth assignments for $\mathbf{P}_1$ and $\mathbf{P}_2$.

Definition 1.6.1 We say that a proposition, P, is *satisfiable* iff it evalates to **true** for *some* truth assignment (taking values in **BOOL**) of the propositional symbols occurring in P and otherwise we say that it is *unsatisfiable*. A proposition, P, is *valid* (or a *tautology*) iff it evaluates to **true** for *all* truth assignments of the propositional symbols occurring in P.

The problem of deciding whether a proposition is satisfiable or not is called the *satisfiability problem* and is sometimes denoted by SAT. The problem of deciding whether a proposition is valid or not is called the *validity problem*. The satisfiability problem is a famous problem in computer science because of its complexity. Try it, solving it is not as easy as you think! In fact, the satisfiability problem turns out to be an *NP-complete* problem, a very important concept that you will learn about in CIS262. The validity problem is also important and it is related to SAT. Indeed, it is easy to see that a proposition, P, is valid iff $\neg P$ is unsatisfiable.

What's the relationship between validity and provability in the system $\mathcal{N}_c^{\Rightarrow,\wedge,\vee,\perp}$ (or $\mathcal{NG}_c^{\Rightarrow,\wedge,\vee,\perp}$)?

Remarkably, in classical logic, validity and provability are equivalent!

In order to prove the above claim, we need to do two things:

(1) Prove that if a proposition, P, if provable in the system $\mathcal{N}_c^{\Rightarrow,\wedge,\vee,\perp}$ (or $\mathcal{NG}_c^{\Rightarrow,\wedge,\vee,\perp}$), then it is valid. This is known as *soundness* or *consistency* (of the proof system).

(2) Prove that if a proposition, P, is valid, then it has a proof in the system $\mathcal{N}_c^{\Rightarrow,\wedge,\vee,\perp}$ (or $\mathcal{NG}_c^{\Rightarrow,\wedge,\vee,\perp}$). This is known as the *completeness* (of the proof system).

In general, it is relatively easy to prove (1) but proving (2) can be quite complicated. In fact, some proof systems are *not* complete with respect to certain semantics. For instance, the proof system for intuitionistic logic, $\mathcal{N}_i^{\Rightarrow,\wedge,\vee,\perp}$ (or $\mathcal{NG}_i^{\Rightarrow,\wedge,\vee,\perp}$), is *not complete* with respect to truth values semantics! As an example, $((P \Rightarrow Q) \Rightarrow P) \Rightarrow P$ (known as *Peirce's law*), is valid but it can be shown that it cannot be proved in intuitionistic logic.

In these notes, we will content ourselves with soundness.

Proposition 1.6.2 *(Soundness of $\mathcal{N}_c^{\Rightarrow,\wedge,\vee,\perp}$ and $\mathcal{NG}_c^{\Rightarrow,\wedge,\vee,\perp}$) If a proposition, P, is provable in the system $\mathcal{N}_c^{\Rightarrow,\wedge,\vee,\perp}$ (or $\mathcal{NG}_c^{\Rightarrow,\wedge,\vee,\perp}$), then it is valid (according to the truth values semantics).*

Sketch of Proof. It is enough to prove that if there is a deduction of a proposition, P, from a set of premises, Γ, then for every truth assignment for which all the propositions in Γ evaluate to **true**, then P evaluates to **true**. However, this is clear for the axioms and every inference rule preserves that property.

Now, if P is provable, a proof of P has an empty set of premises and so P evaluates to **true** for all truth assignments, which means that P is valid. □

Theorem 1.6.3 *(Completeness of $\mathcal{N}_c^{\Rightarrow,\wedge,\vee,\perp}$ and $\mathcal{NG}_c^{\Rightarrow,\wedge,\vee,\perp}$) If a proposition, P, is valid (according to the truth values semantics), then P is provable in the system $\mathcal{N}_c^{\Rightarrow,\wedge,\vee,\perp}$ (or $\mathcal{NG}_c^{\Rightarrow,\wedge,\vee,\perp}$).*

Proofs of completeness for classical logic can be found in van Dalen [44] or Gallier [19] (but for a different proof system).

Soundness (Proposition 1.6.2) has a very useful consequence: In order to prove that a proposition, P, is *not provable*, it is enough to find a truth assignment for which P evaluates to **false**. We say that such a truth assignment is a *counter-example* for P (or that P can be *falsified*). For example, no propositional symbol, $\mathbf{P}_i$, is provable since it is falsified by the truth assignment $\mathbf{P}_i = \textbf{false}$. Note that completeness amounts to the fact that every unprovable formula has a counter-example.

Remark: Truth values semantics is not the right kind of semantics for intuitionistic logic; it is too coarse. A more subtle kind of semantics is required. Among the various semantics for intuitionistic logic, one of the most natural is the notion of *Kripke model*. Then, again, soundness and completeness holds for intuitionistic proof systems (see Section 1.7 and van Dalen [44]).

1.7 Kripke Models for Intuitionistic Logic Soundness and Completeness

In this section, we briefly describe the semantics of intuitionistic propositional logic in terms of Kripke models. This section has been included to quench the thirst of those readers who can't wait to see what kind of decent semantics can be given for intuitionistic propositional logic and it can be safely omitted. We recommend reviewing the material of Section 4.1 before reading this section.

In classical truth values semantics based on $\mathbf{BOOL} = \{\mathbf{true}, \mathbf{false}\}$, we might say that truth is absolute. The idea of Kripke semantics is that there is a set of worlds, W, together with a partial ordering, $\leq$, on W, and that truth depends in which world we are. Furthermore, as we "go up" from a world u to a world v with $u \leq v$, truth "can only increase", that is, whatever is true in world u remains true in world v. Also, the truth of some propositions, such as $P \Rightarrow Q$ or $\neg P$, depends on "future worlds". With this type of semantics, which is no longer absolute, we can capture exactly the essence of intuitionistic logic. We now make these ideas precise.

Definition 1.7.1 A *Kripke model* for intuitionistic propositional logic is a pair, $\mathcal{K} = (W, \varphi)$, where W is a partially ordered (nonempty) set called a *set of worlds* and φ is a function, $\varphi\colon W \to \mathbf{BOOL}^{\mathbf{PS}}$, such that for every $u \in W$, the function $\varphi(u)\colon \mathbf{PS} \to \mathbf{BOOL}$ is an assignment of truth values to the propositional symbols in $\mathbf{PS}$ satisfying the following property: For all $u, v \in W$, for all $\mathbf{P}_i \in \mathbf{PS}$,

$$\text{if } u \leq v \text{ and } \varphi(u)(\mathbf{P}_i) = \mathbf{true}, \quad \text{then } \varphi(v)(\mathbf{P}_i) = \mathbf{true}.$$

As we said in our informal comments, truth can't decrease when we move from a world u to a world v with $u \leq v$ but truth can increase, that is it is possible that $\varphi(u)(\mathbf{P}_i) = \mathbf{false}$ and yet, $\varphi(v)(\mathbf{P}_i) = \mathbf{true}$. We use Kripke models to define the semantics of propositions as follows:

Definition 1.7.2 Given a Kripke model, $\mathcal{K} = (W, \varphi)$, for every $u \in W$ and for every proposition, P, we say that P *is satisfied by* $\mathcal{K}$ *at* u and we write $\varphi(u)(P) = \mathbf{true}$ iff

(a) $\varphi(u)(\mathbf{P}_i) = \mathbf{true}$, if $P = \mathbf{P}_i \in \mathbf{PS}$;

(b) $\varphi(u)(Q) = \mathbf{true}$ and $\varphi(u)(R) = \mathbf{true}$, if $P = Q \wedge R$;

(c) $\varphi(u)(Q) = \mathbf{true}$ or $\varphi(u)(R) = \mathbf{true}$, if $P = Q \vee R$;

(d) For all v such that $u \leq v$, if $\varphi(v)(Q) = \mathbf{true}$, then $\varphi(v)(R) = \mathbf{true}$, if $P = Q \Rightarrow R$;

(e) For all v such that $u \leq v$, $\varphi(v)(Q) = \mathbf{false}$, if $P = \neg Q$.

(f) $\varphi(u)(\perp) = \mathbf{false}$, that is, $\perp$ is not satisfied by $\mathcal{K}$ at u (for any $\mathcal{K}$ and any u).

We say that P is *valid in* $\mathcal{K}$ (or that $\mathcal{K}$ is a *model* of P) iff P is satisfied by $\mathcal{K} = (W, \varphi)$ at u for all $u \in W$ and we say that P is *intuitionistically valid* iff P is valid in every Kripke model, $\mathcal{K}$.

When P is satisfied by $\mathcal{K}$ at u we also say that P *is true at* u *in* $\mathcal{K}$. Note that the truth at $u \in W$ of a proposition of the form $Q \Rightarrow R$ or $\neg Q$ depends on the truth of Q and R at all "future worlds", $v \in W$, with $u \leq v$. Observe that classical truth values semantics corresponds to the special case where W consists of a single element (a single world).

If $W = \{0, 1\}$ ordered so that $0 \leq 1$ and if φ is given by

$$\begin{aligned} \varphi(0)(\mathbf{P}_i) &= \mathbf{false} \\ \varphi(1)(\mathbf{P}_i) &= \mathbf{true}, \end{aligned}$$

then $\mathcal{K} = (W, \varphi)$ is a Kripke structure. The reader should check that the proposition $P = (\mathbf{P}_i \vee \neg\mathbf{P}_i)$ has the value **false** at 0 because $\varphi(0)(\mathbf{P}_i) = \mathbf{false}$ but $\varphi(1)(\mathbf{P}_i) = \mathbf{true}$, so clause (e) fails for $\neg\mathbf{P}_i$ at $u = 0$. Therefore, $P = (\mathbf{P}_i \vee \neg\mathbf{P}_i)$ is not valid in $\mathcal{K}$ and thus, it is not intuitionistically valid. We escaped the classical truth values semantics by using a universe with two worlds. The reader should also check that

$$\begin{aligned} \varphi(u)(\neg\neg P) = \mathbf{true} \quad \text{iff} \quad & \text{for all } v \text{ such that } u \leq v \\ & \text{there some } w \text{ with } v \leq w \text{ so that } \varphi(w)(P) = \mathbf{true}. \end{aligned}$$

This shows that in Krikpe semantics, $\neg\neg P$ is weaker than P, in the sense that
$\varphi(u)(\neg\neg P) = \mathbf{true}$ does not necessarily imply that $\varphi(u)(P) = \mathbf{true}$.

As we said in the previous section, Kripke semantics is a perfect fit to intuitionistic provability in the sense that soundness and completeness hold.

Proposition 1.7.3 *(Soundness of* $\mathcal{N}_i^{\Rightarrow,\wedge,\vee,\perp}$ *and* $\mathcal{NG}_i^{\Rightarrow,\wedge,\vee,\perp}$*) If a proposition, P, is provable in the system* $\mathcal{N}_i^{\Rightarrow,\wedge,\vee,\perp}$ *(or* $\mathcal{NG}_i^{\Rightarrow,\wedge,\vee,\perp}$*), then it is valid in every Kripke model, that is, it is intuitionistically valid.*

Proposition 1.7.3 is not hard to prove. We consider any deduction of a proposition, P, from a set of premises, Γ, and we prove that for every Kripke model, $\mathcal{K} = (W, \varphi)$, for every $u \in W$, if every premise in Γ is satisfied by $\mathcal{K}$ at u, then P is also satisfied by $\mathcal{K}$ at u. This is obvious for the axioms and it is easy to see that the inference rules preserve this property.

Completeness also holds, but it is harder to prove (see van Dalen [44]).

Theorem 1.7.4 *(Completeness of* $\mathcal{N}_i^{\Rightarrow,\wedge,\vee,\perp}$ *and* $\mathcal{NG}_i^{\Rightarrow,\wedge,\vee,\perp}$*) If a proposition, P, is intuitionistically valid, then P is provable in the system* $\mathcal{N}_i^{\Rightarrow,\wedge,\vee,\perp}$ *(or* $\mathcal{NG}_i^{\Rightarrow,\wedge,\vee,\perp}$*).*

Another proof of completeness for a different proof system for propositional intuitionistic logic (a Gentzen-sequent calculus equivalent to $\mathcal{NG}_i^{\Rightarrow,\wedge,\vee,\perp}$) is given in Takeuti [42]. We find this proof more instructive that van Dalen's proof. This proof also shows that if a proposition, P, is not intuitionistically provable, then there is a Kripke model, $\mathcal{K}$, where W is a *finite tree*, in which P is not valid. Such a Kripke model is called a *counter-example* for P.

We now add quantifiers to our language and give the corresponding inference rules.

1.8 Adding Quantifiers; The Proof Systems $\mathcal{N}_c^{\Rightarrow,\wedge,\vee,\forall,\exists,\perp}$, $\mathcal{NG}_c^{\Rightarrow,\wedge,\vee,\forall,\exists,\perp}$

As we mentioned in Section 1.1, atomic propositions may contain variables. The intention is that such variables correspond to arbitrary objects. An example is

$$\text{human}(x) \Rightarrow \text{needs-to-drink}(x).$$

Now, in mathematics, we usually prove universal statements, that is statement that hold for all possible "objects", or existential statement, that is, statement asserting the existence of some object satisfying a given property. As we saw earlier, we assert that every human needs to drink by writing the proposition

$$\forall x(\text{human}(x) \Rightarrow \text{needs-to-drink}(x)).$$

Observe that once the quantifier $\forall$ (pronounced "for all" or "for every") is applied to the variable x, the variable x becomes a place-holder and replacing x by y or any other variable does not change anything. What matters is the locations to which the outer x points to in the inner proposition. We say that x is a *bound variable* (sometimes a "dummy variable").

If we want to assert that some human needs to drink we write

$$\exists x(\text{human}(x) \Rightarrow \text{needs-to-drink}(x));$$

Again, once the quantifier $\exists$ (pronounced "there exists") is applied to the variable x, the variable x becomes a place-holder. However, the intended meaning of the second proposition is very different and weaker than the first. It only asserts the existence of some object satisfying the statement

$$\text{human}(x) \Rightarrow \text{needs-to-drink}(x).$$

Statements may contain variables that are not bound by quantifiers. For example, in

$$\forall y\, \text{parent}(x, y)$$

the variable y is bound but the variable x is not. Here, the intended meaning of parent(x, y) is that x is a parent of y. Variables that are not bound are called *free*. The proposition

$$\forall y \exists x\, \text{parent}(x, y),$$

which contains only bound variables in meant to assert that every y has some parent x. Typically, in mathematics, we only prove statements without free variables. However, statements with free variables may occur during intermediate stages of a proof.

The intuitive meaning of the statement $\forall x P$ is that P holds for all possible objects x and the intuitive meaning of the statement $\exists x P$ is that P holds for some object x. Thus, we see that it would be useful to use symbols to denote various objects. For example, if we want to assert some facts about the "parent" predicate, we may want to introduce some *constant symbols* (for short, constants) such as "Jean", "Mia", *etc.* and write

$$\mathrm{parent}(\mathrm{Jean}, \mathrm{Mia})$$

to assert that Jean is a parent of Mia. Often, we also have to use *function symbols* (or *operators, constructors*), for instance, to write statement about numbers: $+$, $*$, *etc.* Using constant symbols, function symbols and variables, we can form *terms*, such as

$$(x^2 + 1)(3 * y + 2).$$

In addition to function symbols, we also use *predicate symbols*, which are names for atomic properties. We have already seen several examples of predicate symbols: "human", "parent". So, in general, when we try to prove properties of certain classes of objects (people, numbers, strings, graphs, *etc.*), we assume that we have a certain *alphabet* consisting of constant symbols, function symbols and predicate symbols. Using these symbols and an infinite supply of variables (assumed distinct from the variables which we use to label premises) we can form *terms and predicate terms*. We say that we have a *(logical) language*. Using this language, we can write compound statements.

Let us be a little more precise. In a *first-order language*, $\mathbf{L}$, in addition to the logical connectives, $\Rightarrow, \wedge, \vee, \neg, \perp, \forall$ and $\exists$, we have a set, $\mathbf{L}$, of *nonlogical symbols* consisting of

(i) A set $\mathbf{CS}$ of constant symbols, $c_1, c_2, \ldots,$.

(ii) A set $\mathbf{FS}$ of function symbols, $f_1, f_2, \ldots,$. Each function symbol, f, has a *rank*, $n_f \geq 1$, which is the number of arguments of f.

(iii) A set $\mathbf{PS}$ of predicate symbols, $P_1, P_2, \ldots,$. Each predicate symbol, P, has a *rank*, $n_P \geq 0$, which is the number of arguments of P. Predicate symbols of rank 0 are propositional letters, as in earlier sections.

(iv) The equality predicate, $=$, is added to our language when we want to deal with equations.

(v) First-order variables, $t_1, t_2, \ldots$, used to form quantified formulae.

The difference between function symbols and predicate symbols is that function symbols are interpreted as functions defined on a structure (for example, addition, $+$, on $\mathbb{N}$), whereas predicate symbols are interpreted as properties of objects, that is, they take the value **true** or **false**. An example is the language of *Peano arithmetic*, $\mathbf{L} = \{0, S, +, *, =\}$. Here, the intended structure is $\mathbb{N}$, 0 is of course zero, S is interpreted as the function $S(n) = n + 1$, the symbol $+$ is addition, $*$ is multiplication and $=$ is equality.

Using a first-order language, $\mathbf{L}$, we can form terms, predicate terms and formulae. The *terms over* $\mathbf{L}$ are the following expressions:

(i) Every variable, t, is a term;

(ii) Every constant symbol, $c \in \mathbf{CS}$, is a term;

(iii) If $f \in \mathbf{FS}$ is a function symbol taking n arguments and $\tau_1, \ldots, \tau_n$ are terms already constructed, then $f(\tau_1, \ldots, \tau_n)$ is a term.

The *predicate terms over* $\mathbf{L}$ are the following expressions:

(i) If $P \in \mathbf{PS}$ is a predicate symbol taking n arguments and $\tau_1, \ldots, \tau_n$ are terms already constructed, then $P(\tau_1, \ldots, \tau_n)$ is a predicate term. When $n = 0$, the predicate symbol, P, is a predicate term called a propositional letter.

(ii) When we allow the equality predicate, for any two terms τ_1 and τ_2, the expression $\tau_1 = \tau_2$ is a predicate term. It is usually called an *equation*.

The *(first-order) formulae over* $\mathbf{L}$ are the following expressions:

(i) Every predicate term, $P(\tau_1, \ldots, \tau_n)$, is an atomic formula. This includes all propositional letters. We also view $\perp$ (and sometimes $\top$) as an atomic formula.

(ii) When we allow the equality predicate, every equation, $\tau_1 = \tau_2$, is an atomic formula.

(iii) If P and Q are formulae already constructed, then $P \Rightarrow Q$, $P \wedge Q$, $P \vee Q$, $\neg P$ are compound formulae. We treat $P \equiv Q$ as an abbreviation for $(P \Rightarrow Q) \wedge (Q \Rightarrow P)$, as before.

(iv) If P is a formula already constructed and t is any variable, then $\forall t P$ and $\exists t P$ are compound formulae.

All this can be made very precise but this is quite tedious. Our primary goal is to explain the basic rules of logic and not to teach a full-fledged logic course. We hope that our intuitive explanations will suffice and we now come to the heart of the matter, the inference rules for the quantifiers. Once again, for a complete treatment, readers are referred to Gallier [19] van Dalen [44] or Huth and Ryan [31].

Unlike the rules for $\Rightarrow, \vee, \wedge$ and $\perp$, which are rather straightforward, the rules for quantifiers are more subtle due the presence of variables (occurring in terms and predicates). We have to be careful to forbid inferences that would yield "wrong" results and for this we have to be very precise about the way we use free variables. More specifically, we have to exercise care when we make *substitutions* of terms for variables in propositions. For example, say we have the predicate "odd", intended to express that a number is odd. Now, we can substitute the term $(2y+1)^2$ for x in $\mathrm{odd}(x)$ and obtain

$$\mathrm{odd}((2y+1)^2).$$

More generally, if $P(t_1, t_2, \ldots, t_n)$ is a statement containing the free variables $t_1, \ldots, t_n$ and if $\tau_1, \ldots, \tau_n$ are terms, we can form the new statement

$$P[\tau_1/t_1, \ldots, \tau_n/t_n]$$

obtained by substituting the term τ_i for all free occurrences of the variable t_i, for $i = 1, \ldots, n$. By the way, we denote terms by the greek letter τ because we use the letter t for a variable and using t for both variables and terms would be confusing; sorry!

However, if $P(t_1, t_2, \ldots, t_n)$ contains quantifiers, some bad things can happen, namely, some of the variables occurring in some term τ_i may become quantified when τ_i is substituted for t_i. For example, consider

$$\forall x \exists y\, P(x, y, z)$$

which contains the free variable z and substitute the term $x + y$ for z: we get

$$\forall x \exists y\, P(x, y, x + y).$$

We see that the variables x and y occurring in the term $x + y$ become bound variables after substitution. We say that there is a "capture of variables".

This is not what we intended to happen! To fix this problem, we recall that bound variables are really place holders, so they can be renamed without changing anything. Therefore, we can rename the bound variables x and y in $\forall x \exists y\, P(x, y, z)$ to u and v, getting the statement $\forall u \exists v\, P(u, v, z)$ and now, the result of the substitution is

$$\forall u \exists v\, P(u, v, x + y).$$

Again, all this needs to be explained very carefuly but this can be done!

Finally, here are the inference rules for the quantifiers, first stated in a natural deduction style and then in sequent style. It is assumed that we use two disjoint sets of variables for labeling premises $(x, y, \cdots)$ and free variables $(t, u, v, \cdots)$. As we will see, the $\forall$-introduction rule and the $\exists$-elimination rule involve a crucial restriction on the occurrences of certain variables. Remember, *variables are terms*!

Definition 1.8.1 The *inference rules for the quantifiers* are

$\forall$-*introduction*:

$$\cfrac{\cfrac{\Gamma}{P[u/t]}}{\forall tP}$$

Here, u must be a variable that does not occur free in any of the propositions in Γ or in $\forall tP$. The notation $P[u/t]$ stands for the result of substituting u for all free occurrences of t in P. Recall that Γ denotes the set of premises of the above deduction tree so if this tree only has two nodes, then $P[u/t] \in \Gamma$ and t should not occur in P.

$\forall$-*elimination*:

$$\cfrac{\cfrac{\Gamma}{\forall tP}}{P[\tau/t]}$$

Here τ is an arbitrary term and it is assumed that bound variables in P have been renamed so that none of the variables in τ are captured after substitution.

$\exists$-*introduction*:

$$\cfrac{\cfrac{\Gamma}{P[\tau/t]}}{\exists tP}$$

As in $\forall$-elimination, τ is an arbitrary term and the same proviso on bound variables in P applies.

$\exists$-*elimination*:

$$\cfrac{\cfrac{\Gamma}{\exists tP} \qquad \cfrac{\Delta, P[u/t]^x}{C}}{C}\quad x$$

Here, u must be a variable that does not occur free in any of the propositions in Δ, $\exists tP$, or C, and all premises $P[u/t]$ labeled x are discharged.

In the above rules, Γ or Δ may be empty, P, C denote arbitrary propositions constructed from a first-order language, $\mathbf{L}$, and t is *any* variable. The system of *first-order classical logic*, $\mathcal{N}_c^{\Rightarrow,\vee,\wedge,\perp,\forall,\exists}$ is obtained by adding the above rules to the system of propositional classical logic $\mathcal{N}_c^{\Rightarrow,\vee,\wedge,\perp}$. The system of *first-order intuitionistic logic*, $\mathcal{N}_i^{\Rightarrow,\vee,\wedge,\perp,\forall,\exists}$ is obtained by adding the above rules to the system of propositional intuitionistic logic $\mathcal{N}_i^{\Rightarrow,\vee,\wedge,\perp}$.

Using sequents, the quantifier rules in first-order logic are expressed as follows:

Definition 1.8.2 The *inference rules for the quantifiers in Gentzen-sequent style* are

$$\frac{\Gamma \to P[u/t]}{\Gamma \to \forall t P} \quad (\forall\textit{-intro}) \qquad \frac{\Gamma \to \forall t P}{\Gamma \to P[\tau/t]} \quad (\forall\textit{-elim})$$

where in ($\forall$-*intro*), u does not occur free in Γ or $\forall t P$;

$$\frac{\Gamma \to P[\tau/t]}{\Gamma \to \exists t P} \quad (\exists\textit{-intro}) \qquad \frac{\Gamma \to \exists t P \quad z\colon P[u/t], \Gamma \to C}{\Gamma \to C} \quad (\exists\textit{-elim})$$

where in ($\exists$-*elim*), u does not occur free in Γ, $\exists t P$, or C. Again, t is *any* variable.

The variable u is called the *eigenvariable* of the inference. The systems $\mathcal{NG}_c^{\Rightarrow,\vee,\wedge,\perp,\forall,\exists}$ and $\mathcal{NG}_i^{\Rightarrow,\vee,\wedge,\perp,\forall,\exists}$ are defined from the systems $\mathcal{NG}_c^{\Rightarrow,\vee,\wedge,\perp}$ and $\mathcal{NG}_i^{\Rightarrow,\vee,\wedge,\perp}$, respectively, by adding the above rules.

When we say that a proposition, P, is *provable from* Γ, we mean that we can construct a proof tree whose conclusion is P and whose set of premises is Γ, in one of the systems $\mathcal{N}_c^{\Rightarrow,\wedge,\vee,\perp,\forall,\exists}$ or $\mathcal{NG}_c^{\Rightarrow,\wedge,\vee,\perp,\forall,\exists}$. Therefore, as in propositional logic, when we use the word "provable" unqualified, we mean provable in *classical logic*. Otherwise, we say *intuitionistically provable* .

A first look at the above rules shows that universal formulae, $\forall t P$, behave somewhat like infinite conjunctions and that existential formulae, $\exists t P$, behave somewhat like infinite disjunctions.

The $\forall$-introduction rule looks a little strange but the idea behind it is actually very simple: Since u is totally unconstrained, if $P[u/t]$ is provable (from Γ), then intuitively $P[u/t]$ holds of any arbitrary object, and so, the statement $\forall t P$ should also be provable (from Γ). Note that the tree

$$\frac{P[u/t]}{\forall t P}$$

is generally an illegal deduction because it has the single premise, $P[u/t]$, and u occurs in $P[u/t]$ unless t does not occur in P.

The meaning of the $\forall$-elimination is that if $\forall t P$ is provable (from Γ), then P holds for all objects and so, in particular for the object denoted by the term τ, i.e., $P[\tau/t]$ should be provable (from Γ).

The $\exists$-introduction rule is dual to the $\forall$-elimination rule. If $P[\tau/t]$ is provable (from Γ), this means that the object denoted by τ satisfies P, so $\exists t P$ should be provable (this latter formula asserts the existence of some object satisfying P, and τ is such an object).

The $\exists$-elimination rule is reminiscent of the $\vee$-elimination rule and is a little more tricky. It goes as follows: Suppose that we proved $\exists t P$ (from Γ). Moreover, suppose that for every possible case, $P[u/t]$, we were able to prove C (from Γ). Then, as we have "exhausted" all possible cases and as we know from the provability of $\exists tP$ that some case must hold, we can conclude that C is provable (from Γ) without using $P[u/t]$ as a premise.

Like the $\vee$-elimination rule, the $\exists$-elimination rule is not very constructive. It allows making a conclusion (C) by considering alternatives without knowing which one actually occurs.

Remark: Anagolously to disjunction, in (first-order) intuitionistic logic, if an existential statement $\exists tP$ is provable, then from any proof of $\exists tP$, some term, τ, can be extracted so that $P[\tau/t]$ is provable. Such a term, τ, is called a *witness*. The witness property is not easy to prove. It follows from the fact that intuitionistic proofs have a normal form (see Section 1.9). However, no such property holds in classical logic (for instance, see the a^b rational with a, b irrational example revisited below).

Here is an example of a proof in the system $\mathcal{N}_c^{\Rightarrow,\vee,\wedge,\perp,\forall,\exists}$ (actually, in $\mathcal{N}_i^{\Rightarrow,\vee,\wedge,\perp,\forall,\exists}$) of the formula $\forall t(P \wedge Q) \Rightarrow \forall tP \wedge \forall tQ$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\forall t(P \wedge Q)^x}{P[u/t] \wedge Q[u/t]}}{P[u/t]}}{\forall tP} \qquad \dfrac{\dfrac{\dfrac{\forall t(P \wedge Q)^x}{P[u/t] \wedge Q[u/t]}}{Q[u/t]}}{\forall tQ}}{\dfrac{\forall tP \wedge \forall tQ}{\forall t(P \wedge Q) \Rightarrow \forall tP \wedge \forall tQ}\ x}$$

In the above proof, u is a new variable, i.e., a variable that does not occur free in P or Q. We also have used some basic properties of substitutions such as:

$$\begin{array}{rcl}
(P \wedge Q)[\tau/t] & = & P[\tau/t] \wedge Q[\tau/t] \\
(P \vee Q)[\tau/t] & = & P[\tau/t] \vee Q[\tau/t] \\
(P \Rightarrow Q)[\tau/t] & = & P[\tau/t] \Rightarrow Q[\tau/t] \\
(\neg P)[\tau/t] & = & \neg P[\tau/t] \\
(\forall sP)[\tau/t] & = & \forall sP[\tau/t] \\
(\exists sP)[\tau/t] & = & \exists sP[\tau/t],
\end{array}$$

for any term, τ, such that no variable in τ is captured during the substitution (in particular, in the last two cases, the variable s does not occur in τ).

The reader should show that $\forall tP \wedge \forall tQ \Rightarrow \forall t(P \wedge Q)$ is also provable in $\mathcal{N}_i^{\Rightarrow,\vee,\wedge,\perp,\forall,\exists}$. However, in general, one can't just replace $\forall$ by $\exists$ (or $\wedge$ by $\vee$) and still obtain provable statements. For example, $\exists tP \wedge \exists tQ \Rightarrow \exists t(P \wedge Q)$ is not provable at all!

Here are some useful equivalences involving quantifiers. The first two are analogous to the de Morgan laws for $\wedge$ and $\vee$.

Proposition 1.8.3 *The following equivalences are provable in classical first-order logic:*

$$\begin{aligned}\neg\forall t P &\equiv \exists t \neg P\\ \neg\exists t P &\equiv \forall t \neg P\\ \forall t(P \wedge Q) &\equiv \forall t P \wedge \forall t Q\\ \exists t(P \vee Q) &\equiv \exists t P \vee \exists t Q.\end{aligned}$$

In fact, the last three and $\exists t\neg P \Rightarrow \neg\forall t P$ are provable intuitionistically. Moreover, the propositions $\exists t(P\wedge Q) \Rightarrow \exists t P \wedge \exists t Q$ and $\forall t P \vee \forall t Q \Rightarrow \forall t(P \vee Q)$ are provable in intuitionistic first-order logic (and thus, also in classical first-order logic).

Proof. Left as an exercise to the reader. □

Remark: We can illustrate, again, the fact that classical logic allows for non-constructive proofs by reexamining the example at the end of Section 1.3. There, we proved that if $\sqrt{2}^{\sqrt{2}}$ is rational, then $a = \sqrt{2}$ and $b = \sqrt{2}$ are both irrational numbers such that a^b is rational and if $\sqrt{2}^{\sqrt{2}}$ is irrational then $a = \sqrt{2}^{\sqrt{2}}$ and $b = \sqrt{2}$ are both irrational numbers such that a^b is rational. By $\exists$-introduction, we deduce that if $\sqrt{2}^{\sqrt{2}}$ is rational then there exist some irrational numbers a, b so that a^b is rational and if $\sqrt{2}^{\sqrt{2}}$ is irrational then there exist some irrational numbers a, b so that a^b is rational. In classical logic, as $P \vee \neg P$ is provable, by $\vee$-elimination, we just proved that there exist some irrational numbers a and b so that a^b is rational.

However, this argument does not give us explicitly numbers a and b with the required properties! It only tells us that such numbers must exist. Now, it turns out that $\sqrt{2}^{\sqrt{2}}$ is indeed irrational (this follows from the Gel'fond-Schneider Theorem, a hard theorem in number theory). Furthermore, there are also simpler explicit solutions such as $a = \sqrt{2}$ and $b = \log_2 9$, as the reader should check!

We conclude this section by giving an example of a "wrong proof". Here is an example in which the $\forall$-introduction rule is applied illegally, and thus, yields a statement which is actually false (not provable). In the incorrect "proof" below, P is an atomic predicate symbol taking two arguments (for example, "parent") and 0 is a constant denoting zero:

$$\cfrac{\cfrac{\cfrac{\cfrac{P(t,0)^x}{\forall t P(t,0)}\quad \text{illegal step!}}{P(t,0) \Rightarrow \forall t P(t,0)}\quad x}{\forall t(P(t,0) \Rightarrow \forall t P(t,0))}}{P(0,0) \Rightarrow \forall t P(t,0)}$$

The problem is that the variable t occurs free in the premise $P[t/t, 0] = P(t, 0)$ and therefore, the application of the $\forall$-introduction rule in the first step is illegal. However, note that this premise is discharged in the second step and so, the application of the $\forall$-introduction rule in the third step is legal. The (false) conclusion of this faulty proof is that $P(0,0) \Rightarrow \forall t P(t,0)$ is provable. Indeed, there are plenty of properties such that the fact that the single instance, $P(0,0)$, holds does not imply that $P(t,0)$ holds for all t.

Remark: The above example shows why it is desirable to have premises that are universally quantified. A premise of the form $\forall t P$ can be instantiated to $P[u/t]$, using $\forall$-elimination, where u is a brand new variable. Later on, it may be possible to use $\forall$-introduction without running into trouble with free occurrences of u in the premises. But we still have to be very careful when we use $\forall$-introduction or $\exists$-elimination.

Before concluding this section, let us give a few more examples of proofs using the rules for the quantifiers. First, let us prove that

$$\forall t P \equiv \forall u P[u/t],$$

where u is any variable not free in $\forall t P$ and such that u is not captured during the substitution. This rule allows us to rename bound variables (under very mild conditions). We have the proofs

$$\dfrac{\dfrac{\dfrac{(\forall t P)^{\alpha}}{P[u/t]}}{\forall u P[u/t]}}{\forall t P \Rightarrow \forall u P[u/t]}\quad \alpha$$

and

$$\dfrac{\dfrac{\dfrac{(\forall u P[u/t])^{\alpha}}{P[u/t]}}{\forall t P}}{\forall u P[u/t] \Rightarrow \forall t P}\quad \alpha$$

Now, we give a proof (intuitionistic) of

$$\exists t (P \Rightarrow Q) \Rightarrow (\forall t P \Rightarrow Q),$$

where t does not occur (free or bound) in Q.

$$\dfrac{\dfrac{\dfrac{(\exists t(P \Rightarrow Q))^z \qquad \dfrac{(P[u/t] \Rightarrow Q)^x \qquad \dfrac{(\forall t P)^y}{P[u/t]}}{Q}}{Q}\ x}{\forall t P \Rightarrow Q}\ y}{\exists t(P \Rightarrow Q) \Rightarrow (\forall t P \Rightarrow Q)}\ z$$

In the above proof, u is a new variable that does not occur in Q, $\forall t P$, or $\exists t(P \Rightarrow Q)$. Since t does not occur in Q, we have

$$(P \Rightarrow Q)[u/t] = P[u/t] \Rightarrow Q.$$

The converse requires (RAA) and is a bit more complicated. To conclude, we give a proof (intuitionistic) of

$$(\forall t P \vee Q) \Rightarrow \forall t(P \vee Q),$$

where t does not occur (free or bound) in Q.

$$\dfrac{\dfrac{(\forall t P \vee Q)^z \qquad \dfrac{\dfrac{\dfrac{(\forall t P)^x}{P[u/t]}}{P[u/t] \vee Q}}{\forall t(P \vee Q)} \qquad \dfrac{\dfrac{Q^y}{P[u/t] \vee Q}}{\forall t(P \vee Q)}}{\forall t(P \vee Q)}\ x,y}{(\forall t P \vee Q) \Rightarrow \forall t(P \vee Q)}\ z$$

In the above proof, u is a new variable that does not occur in $\forall t P$ or Q. Since t does not occur in Q, we have

$$(P \vee Q)[u/t] = P[u/t] \vee Q.$$

The converse requires (RAA).

Obviously, every first-order formula that is provable intuitionistically is also provable classically and we know that there are formulae that are provable classically but *not* provable intuitionistically. Therefore, it appears that classical logic is more general than intuitionistic logic. However, this not not quite so because there is a way of interpreting classical logic into intuitionistic logic. To be more precise, every classical formula, A, can be translated into a formula, A^*, where A^* is classically equivalent to A and A is provable classically iff A^* is provable intuitionistically. Various translations are known, all based on a “trick” involving double-negation (This is because $\neg\neg\neg A$ and $\neg A$ are intuitionistically equivalent).

Translations were given Kolmogorov (1925), Gödel (1933) and Gentzen (1933). For example, Gödel used the following translation:

$$\begin{aligned}
A^* &= \neg\neg A, \quad \text{if } A \text{ is atomic},\\
(\neg A)^* &= \neg A^*,\\
(A \wedge B)^* &= (A^* \wedge B^*),\\
(A \Rightarrow B)^* &= \neg(A^* \wedge \neg B^*),\\
(A \vee B)^* &= \neg(\neg A^* \wedge \neg B^*),\\
(\forall x A)^* &= \forall x A^*,\\
(\exists x A)^* &= \neg\forall x \neg A^*.
\end{aligned}$$

Actually, if we restrict our attention to propositions (that is, formulae without quantifiers), a theorem of Glivenko (1929) states that if a proposition, A, is provable classically, then $\neg\neg A$ is provable intuitionistically. In view of these results, the proponents of intuitionistic logic claim that classical logic is really a special case of intuitionistic logic! However, the above translations have some undesirable properties, as noticed by Girard. For more details on all this, see Gallier [17].

Several times in this Chapter, we have claimed that certain propositions are not provable in some logical system. What kind of reasoning do we use to validate such claims? In the next section, we briefly address this question as well as related ones.

1.9 Decision Procedures, Proof Normalization, Counter-Examples, Theories, etc.

In the previous sections, we saw how the rules of mathematical reasoning can be formalized in various natural deduction systems and we defined a precise notion of proof. We observed that finding a proof for a given proposition was not a simple matter, nor was it to acertain that a proposition is unprovable. Thus, it is natural to ask the following question:

The Decision Problem: Is there a general procedure which takes any arbitrary proposition, P, as input, always terminates in a finite number of steps, and tells us whether P is provable or not.

Clearly, it would be very nice if such a procedure existed, especially if it also produced a proof of P when P is provable.

Unfortunately, for rich enough languages, such as first-order logic, it is impossible to find such a procedure. This deep result known as the *undecidability of the decision problem* or *Church's Theorem* was proved by A. Church in 1936 (Actually, Church proved the undecidability of the validity problem but, by Gödel's completeness Theorem, validity and provability are equivalent).

Proving Church's Theorem is hard and a lot of work. One needs to develop a good deal of what is called the *theory of computation*. This involves defining models of computation such as *Turing machines* and proving other deeps results such as the *undecidability of the halting problem* and the *undecidability of the Post Correspondence Problem*, among other things. Some of this material is covered in CIS262, so be patient and your curiosity will be satisfied!

So, our hopes to find a "universal theorem prover" are crushed. However, if we restrict ourselves to propositional logic, classical or intuitionistic, it turns out that procedures solving the decision problem do exist and they even produce a proof of the input proposition when that proposition is provable.

Unfortunately, proving that such procedures exist and are correct in the propositional case is rather difficult, especially for intuitionistic logic. The difficulties have a lot to do with our choice of a natural deduction system. Indeed, even for the system $\mathcal{N}_m^{\Rightarrow}$ (or $\mathcal{NG}_m^{\Rightarrow}$), provable propositions may have infinitely many proofs. This makes the search process impossible; when do we know how to stop, especially if a proposition is not provable! The problem is that proofs may contain redundancies (Gentzen said "detours"). A typical example of redundancy is an elimination immediately follows an introduction, as in the following example in which $\mathcal{D}_1$ denotes a deduction with conclusion $\Gamma, x\colon A \to B$ and $\mathcal{D}_2$ denotes a deduction with conclusion $\Gamma \to A$.

$$\dfrac{\dfrac{\begin{array}{c}\mathcal{D}_1\\ \Gamma, x\colon A \to B\end{array}}{\Gamma \to A \Rightarrow B} \qquad \begin{array}{c}\mathcal{D}_2\\ \Gamma \to A\end{array}}{\Gamma \to B}$$

Intuitively, it should be possible to construct a deduction for $\Gamma \to B$ from the two deductions $\mathcal{D}_1$ and $\mathcal{D}_2$ without using at all the hypothesis $x\colon A$. This is indeed the case. If we look closely at the deduction $\mathcal{D}_1$, from the shape of the inference rules, assumptions are never created, and the leaves must be labeled with expressions of the form $\Gamma', \Delta, x\colon A, y\colon C \to C$ or $\Gamma, \Delta, x\colon A \to A$, where $y \neq x$ and either $\Gamma = \Gamma'$ or $\Gamma = \Gamma', y\colon C$. We can form a new deduction for $\Gamma \to B$ as follows: in $\mathcal{D}_1$, wherever a leaf of the form $\Gamma, \Delta, x\colon A \to A$ occurs, replace it by the deduction obtained from $\mathcal{D}_2$ by adding Δ to the premise of each sequent in $\mathcal{D}_2$. Actually, one should be careful to first make a fresh copy of $\mathcal{D}_2$ by renaming all the variables so that clashes with variables in $\mathcal{D}_1$ are avoided. Finally, delete the assumption $x\colon A$ from the premise of every sequent in the resulting proof. The resulting deduction is obtained by a kind of substitution and may be denoted as $\mathcal{D}_1[\mathcal{D}_2/x]$, with some minor abuse of notation. Note that the assumptions $x\colon A$ occurring in the leaves of the form $\Gamma', \Delta, x\colon A, y\colon C \to C$ were never used anyway. The step which consists in transforming the above redundant proof figure into the deduction $\mathcal{D}_1[\mathcal{D}_2/x]$ is called a *reduction step* or *normalization step*.

The idea of *proof normalization* goes back to Gentzen ([21], 1935). Gentzen noted that (formal) proofs can contain redundancies, or "detours", and that most complications in the analysis of proofs are due to these redundancies. Thus, Gentzen had the idea that the analysis of proofs would be simplified if it was possible to show that every proof can be converted to an equivalent irredundant proof, a proof in normal form. Gentzen proved a technical result to that effect, the "cut-elimination theorem", for a sequent-calculus formulation of first-order logic [21]. Cut-free proofs are direct, in the sense that they never use auxiliary lemmas via the cut rule.

Remark: It is important to note that Gentzen's result gives a particular algorithm to produce a proof in normal form. Thus, we know that every proof can be reduced to some normal form using a specific strategy, but there may be more than one normal form, and certain normalization strategies may not terminate.

About thirty years later, Prawitz ([36], 1965) reconsidered the issue of proof normalization, but in the framework of natural deduction rather than the framework of sequent calculi.[1] Prawitz explained very clearly what redundancies are in systems of natural deduction, and he proved that every proof can be reduced to a normal form. Furthermore, this normal form is unique. A few years later, Prawitz ([37], 1971) showed that in fact, every reduction sequence terminates, a property also called *strong normalization*.

A remarkable connection between proof normalization and the notion of computation must also be mentioned. Curry (1958) made the remarkably insightful observation that certain typed combinators can be viewed as representations of proofs (in a Hilbert system) of certain propositions (See in Curry and Feys [13] (1958), Chapter 9E, pages 312-315.) Building up on this observation, Howard ([30], 1969) described a general correspondence between propositions and types, proofs in natural deduction and certain typed λ-terms, and proof normalization and β-reduction. (The simply-typed-λ-calculus was invented by Church, 1940). This correspondence, usually referred to as the *Curry/Howard isomorphism* or *formulae–as–types principle*, is fundamental and very fruitful.

The Curry/Howard isomorphism establishes a deep correspondence between the notion of proof and the notion of computation. Furthermore, and this is the deepest aspect of the Curry/Howard isomorphism, proof normalization corresponds to term reduction in the λ-calculus associated with the proof system. To make the story short, the correspondence between proofs in intuitionistic logic and typed λ-terms on one-hand and between proof normalization and β-conversion on the other hand can be used to translate results about typed λ-terms into results about proofs in intuitionistic logic. By the way, some aspects of the Curry/Howard isomorphism are covered in CIS500.

In summary, using either some suitable intuitionistic sequent calculi and Gentzen's cut elimination theorem or some suitable typed λ-calculi and (strong) normalization results about them, it is possible to prove that there is a decision procedure for propositional intuitionistic logic. However, it can also be shown that the time-complexity of any such procedure is very high. Here, we are alluding to *complexity theory*, another active area of computer

[1] This is somewhat ironical, since Gentzen began his investigations using a natural deduction system, but decided to switch to sequent calculi (known as Gentzen systems!) for technical reasons.

science. You will learn about some basic and fundamental aspects of this theory in CIS262 when you learn about the two problems P and NP.

Readers who wish to learn more about these topics can read my two survey papers Gallier [18] (on the Correspondence Between Proofs and λ-Terms) and Gallier [17] (A Tutorial on Proof Systems and Typed λ-Calculi), both available on the web site

http://www.cis.upenn.edu/˜jean/gbooks/logic.html

and the excellent introduction to proof theory by Troelstra and Schwichtenberg [43].

Anybody who really wants to understand logic should of course take a look at Kleene [32] (the famous "I.M."), but this is not recommended to beginners!

Let us return to the question of deciding whether a proposition is not provable. To simplify the discussion, let us restrict our attention to propositional classical logic. So far, we have presented a very *proof-theoretic* view of logic, that is, a view based on the notion of provability as opposed to a more *semantic* view of based on the notions of truth and models. A possible excuse for our bias is that, as Peter Andrews (from CMU) puts it, "truth is elusive". Therefore, it is simpler to understand what truth is in terms of the more "mechanical" notion of provability. (Peter Andrews even gave the subtitle

To Truth Through Proof

to his logic book Andrews [1]!)

However, mathematicians are not mechanical theorem provers (even if they prove lots of stuff)! Indeed, mathematicians almost always think of the objects they deal with (functions, curves, surfaces, groups, rings, *etc.*) as rather concrete objects (even if they may not seem concrete to the uninitiated) and not as abstract entities solely characterized by arcane axioms.

It is indeed natural and fruitful to try to interpret formal statements semantically. For propositional classical logic, this can be done quite easily if we interpret atomic propositional letters using the truth values **true** and **false**, as explained in Section 1.6. Then, the crucial point that *every provable proposition (say in $\mathcal{NG}_c^{\Rightarrow,\vee,\wedge,\perp}$) has the value* **true** *no matter how we assign truth values to the letters in our proposition*. In this case, we say that P is *valid*.

The fact that provability implies validity is called *soundness* or *consistency* of the proof system. The soundness of the proof system $\mathcal{NG}_c^{\Rightarrow,\vee,\wedge,\perp}$ is easy to prove, as sketched in Section 1.6.

We now have a method to show that a proposition, P, is not provable: Find some truth assignment that makes P **false**.

Such an assignment falsifying P is called a *counter-example*. If P has a counter-example, then it can't be provable because if it were, then by soundness it would be **true** for all possible truth assignments.

But now, another question comes up: If a proposition is not provable, can we always find a counter-example for it. Equivalently, *is every valid proposition provable*? If every valid proposition is provable, we say that our proof system is *complete* (this is the *completeness* of our system).

The system $\mathcal{NG}_c^{\Rightarrow,\vee,\wedge,\perp}$ is indeed complete. In fact, *all* the classical systems that we have discussed are sound and complete. Completeness is usually a lot harder to prove than soundness. For first-order classical logic, this is known as *Gödel's completeness Theorem* (1929). Again, we refer our readers to Gallier [19] van Dalen [44] or or Huth and Ryan [31] for a thorough discussion of these matters. In the first-order case, one has to define *first-order structures* (or *first-order models*).

What about intuitionistic logic?

Well, one has to come up with a richer notion of semantics because it is no longer true that if a proposition is valid (in the sense of our two-valued semantics using **true**, **false**), then it is provable. Several semantics have been given for intuitionistic logic. In our opinion, the most natural is the notion of *Kripke model*, presented in Section 1.7. Then, again, soundness and completeness holds for intuitionistic proof systems, even in the first-order case (see Section 1.7 and van Dalen [44]).

In summary, semantic models can be use to provide *counter-examples* of unprovable propositions. This is a quick method to establish that a proposition is not provable.

The way we presented deduction trees and proof trees may have given our readers the impression that the set of premises, Γ, was just an auxiliary notion. Indeed, in all of our examples, Γ ends up being empty! However, nonempty Γ's are crucially needed if we want to develop theories about various kinds of structures and objects, such as the natural numbers, groups, rings, fields, trees, graphs, sets, *etc.* Indeed, we need to make definitions about the objects we want to study and we need to state some axioms asserting the main properties of these objects. We do this by putting these definitions and axioms in Γ. Actually, we have to allow Γ to be infinite but we still require that our deduction trees are finite; they can only use finitely many of the propositions in Γ. We are then interested in all propositions, P, such that $\Delta \to P$ is provable, where Δ is any finite subset of Γ; the set of all such P's is called a *theory*. Of course we have the usual problem of consistency: If we are not careful, our theory may be inconsistent, i.e., it may consist of all propositions.

Let us give two examples of theories.

Our first example is the *theory of equality*. Indeed, our readers may have noticed that we have avoided to deal with the equality relation. In practice, we can't do that.

Given a language, $\mathbf{L}$, with a given supply of constant, function and predicate symbols, the theory of equality consists of the following formulae taken as axioms:

$$\begin{aligned}
&\forall(x = x)\\
&\forall x_1 \cdots \forall x_n \forall y_1 \cdots \forall y_n [(x_1 = y_1 \wedge \cdots \wedge x_n = y_n) \Rightarrow f(x_1, \ldots, x_n) = f(y_1, \ldots, y_n)]\\
&\forall x_1 \cdots \forall x_n \forall y_1 \cdots \forall y_n [(x_1 = y_1 \wedge \cdots \wedge x_n = y_n) \wedge P(x_1, \ldots, x_n) \Rightarrow P(y_1, \ldots, y_n)],
\end{aligned}$$

for all function symbols (of n arguments) and all predicate symbols (of n arguments), including the equality predicate, $=$, itself.

It is not immediately clear from the above axioms that $=$ is reflexive and transitive but this can shown easily.

Our second example is the first-order theory of the natural numbers known as *Peano's arithmetic*.

Here, we have the constant 0 (zero), the unary function symbol S (for successor function; the intended meaning is $S(n) = n + 1$) and the binary function symbols $+$ (for addition) and $*$ (for multiplication). In addition to the axioms for the theory of equality we have the following axioms:

$$
\begin{aligned}
&\forall x \neg(S(x) = 0)\\
&\forall x \forall y (S(x) = S(y) \Rightarrow x = y)\\
&\forall x \forall y (x + 0 = x)\\
&\forall x \forall y (x + S(y) = S(x + y))\\
&\forall x \forall y (x * 0 = 0)\\
&\forall x \forall y (x * S(y) = x * y + x)\\
&[A(0) \wedge \forall x (A(x) \Rightarrow A(S(x)))] \Rightarrow \forall n A(n),
\end{aligned}
$$

where A is any first-order formula with one free variable. This last axiom is the *induction axiom*. Observe how $+$ and $*$ are defined recursively in terms of 0 and S and that there are infinitely many induction axioms (countably many).

Many properties that hold for the natural numbers (i.e., are true when the symbols $0, S, +, *$ have their usual interpretation and all variables range over the natural numbers) can be proved in this theory (Peano's arithmetic), but not all! This is another very famous result of Gödel known as *Gödel's incompleteness Theorem* (1931). However, the topic of incompleteness is definitely oustside the scope of this course, so we will not say anymore about it. Another very interesting theory is *set theory*. There are a number of axiomatizations of set theory and we will discuss one of them (ZF) very briefly in the next section.

We close this section by repeating something we said ealier: There isn't just one logic but instead, *many* logics. In addition to classical and intuitionistic logic (propositional and first-order), there are: modal logics, higher-order logics and *linear logic*, a logic due to Jean-Yves Girard, attempting to unify classical and intuitionistic logic (among other goals). An excellent introduction to these logics can be found in Troelstra and Schwichtenberg [43]. We warn our readers that most presentations of linear logic are (very) difficult to follow. This is definitely true of Girard's seminal paper [23]. A more approachable version can be found in Girard, Lafont and Taylor [22], but most readers will still wonder what hit them when they attempt to read it.

In computer science, there is also *dynamic logic*, used to prove properties of programs and *temporal logic* and its variants (originally invented by A. Pnueli), to prove properties of real-time systems. So, logic is alive and well! Also, take a look at CIS482!

1.10 Basics Concepts of Set Theory

Having learned some fundamental notions of logic, it is now a good place before proceeding to more interesting things, such as functions and relations, to go through a very quick review of some basic concepts of set theory. This section will take the very "naive" point of view that a set is a collection of objects, the collection being regarded as a single object. Having first-order logic at our disposal, we could formalize set theory very rigorously in terms of axioms. This was done by Zermelo first (1908) and in a more satisfactory form by Zermelo and Fraenkel in 1921, in a theory known as the "Zermelo-Fraenkel" (ZF) axioms. Another axiomatization was given by John von Neumann in 1925 and later improved by Bernays in 1937. A modification of Bernay's axioms was used by Kurt Gödel in 1940. This approach is now known as "von Neumann-Bernays" (VNB) or "Gödel-Bernays" (GB) set theory. There are many books that give an axiomatic presentation of set theory. Among them, we recommend Enderton [15], which we find remarkably clear and elegant, Suppes [41] (a little more advanced) and Halmos [28], a classic (at a more elementary level).

However, it must be said that set theory was first created by Georg Cantor (1845-1918) between 1871 and 1879. However, Cantor's work was not unanimously well received by all mathematicians. Cantor regarded infinite objects as objects to be treated in much the same way as finite sets, a point of view that was shocking to a number of very prominent mathematicians who bitterly attacked him (among them, the powerful Kronecker). Also, it turns out that some paradoxes in set theory popped up in the early 1900, in particular, Russell's paradox. Russell's paradox (found by Russell in 1902) has to to with the

"set of all sets that are not members of themselves"

which we denote by

$$R = \{x \mid x \notin x\}.$$

(In general, the notation $\{x \mid P\}$ stand for the set of all objects satisfying the property P.)

Now, classically, either $R \in R$ or $R \notin R$. However, if $R \in R$, then the definition of R says that $R \notin R$; if $R \notin R$, then again, the definition of R says that $R \in R$!

So, we have a contradiction and the existence of such a set is a paradox. The problem is that we are allowing a property (here, $P(x) = x \notin x$), which is "too wild" and circular in nature. As we will see, the way out, as found by Zermelo, is to place a restriction on the property P and to also make sure that P picks out elements from some already given set (see the Subset Axioms below).

The apparition of these paradoxes prompted mathematicians, with Hilbert among its leaders, to put set theory on firmer grounds. This was achieved by Zermelo, Fraenkel, von Neumann, Bernays and Gödel, to only name the major players.

In what follows, we are assuming that we are working in classical logic. We will introduce various operations on sets using definitions involving the logical connectives $\wedge$, $\vee$, $\neg$, $\forall$ and $\exists$. In order to ensure the existence of some of these sets requires some of the axioms of set theory, but we will be rather casual about that.

Given a set, A, we write that some object, a, is an element of (belongs to) the set A as

$$a \in A$$

and that a is not an element of A (does not belong to A) as

$$a \notin A.$$

When are two sets A and B equal? This corresponds to the first axiom of set theory, called

Extensionality Axiom

Two sets A and B are equal iff they have exactly the same elements, that is

$$\forall x(x \in A \Rightarrow x \in B) \wedge \forall x(x \in B \Rightarrow x \in A).$$

The above says: Every element of A is an element of B and conversely.

There is a special set having no elements at all, the *empty set*, denoted $\emptyset$. This is the

Empty Set Axiom

There is a set having no members. This set is denoted $\emptyset$ and it is characterized by the property

$$\forall x(x \notin \emptyset).$$

Remark: Beginners often wonder whether there is more than one empty set. For example, is the empty set of professors distinct from the empty set of potatoes?

The answer is, by the extensionality axiom, there is only *one* empty set!

Given any two objects a and b, we can form the set $\{a, b\}$ containing exactly these two objects. Amazingly enough, this must also be an axiom:

Pairing Axiom

Given any two objects a and b (think sets), there is a set, $\{a, b\}$, having as members just a and b.

Observe that if a and b are identical, then we have the set $\{a, a\}$, which is denoted by $\{a\}$ and is called a *singleton set* (this set has a as its only element).

To form bigger sets, we use the union operation. This too requires an axiom.

Union Axiom (Version 1)