2 $\mathcal{A}$: the logic of All p are q

This book studies a variety of logical systems, starting in this chapter with what must be the simplest logical system "of all", and then adding more and more features to it. The logical language has only one kind of sentence: All p are q. That's it. No "if"s, "and"s, or "but"s. So the *syntax* here is going to be very small. We will then see a *semantics* consisting of a short list of definitions that explain the meaning of the language in *models* and some related points. And then we turn to a *proof system*.

The language and logical system in this chapter are so small that you might think that the chapter is boring or uninteresting. We encourage you to take the chapter as something to master and find interesting even though it might be simple. We also hope that you will return to it as you read further in the book, since we want to use the chapter as a first example of the many logical systems that we shall see in later chapters.

2.1 Syntax and semantics

All of our logical systems have a *syntax* and a *semantics*. We will discuss what these terms mean later, but for now, let us say that syntax is about the form of sentences, and semantics is about meaning.

For the syntax of our first system, we start with a collection $\mathbf{P}$ of *atoms*[1] (for nouns). The elements of $\mathbf{P}$ may be anything, and $\mathbf{P}$ might be finite or infinite. We write the atoms as p, q, $\ldots$; occasionally we use subscripts or other devices.

Definition 2.1: the language $\mathcal{A}$

We take as *sentences* the expressions

$$\text{All } p \text{ are } q$$

where p and q are any atoms in $\mathbf{P}$. There is nothing else in the language. We call this language $\mathcal{A}$.

$\mathcal{A}$ stands for "all." Note that $\mathcal{A}$ is really a family of languages, one for each set $\mathbf{P}$ of atoms at the outset. This dependence on primitives is true for practically all logical systems. For most of the work in this book, we suppress mention of the primitive syntactic items because it makes the notation lighter and because we rarely need to call attention to the primitives in the first place.

[1] In Section 3.1 and in Chapter 5 and onward, we'll also have *binary atoms* to represent transitive verbs and other things. So what we are calling *atoms* here will later be re-named to *unary* atoms.

Definition 2.2: models

A model $\mathcal{M}$ for this language $\mathcal{A}$ is a structure

$$\mathcal{M} \quad = \quad (M, [\![\ ]\!])$$

consisting of a set M, together with an *interpretation function*

$$[\![\ ]\!] : \mathbf{P} \to \mathcal{P}(M)$$

which assigns to each atom p a subset $[\![p]\!] \subseteq M$. We read $[\![p]\!]$ as "the meaning of p in the model $\mathcal{M}$."
M is called the *universe of* $\mathcal{M}$. Frequently we name a structure by its universe, so we would

Definition 2.3: truth in a model

The main semantic definition is *truth in a model*:

$$\mathcal{M} \models \textsf{All } p \textsf{ are } q \qquad \text{iff} \qquad [\![p]\!] \subseteq [\![q]\!].$$

We read this notation in various ways, such as $\mathcal{M}$ *satisfies* All p are q, or All p are q is *true in* $\mathcal{M}$.
If $\mathcal{M}$ does not satisfy All p are q, we would write $\mathcal{M} \not\models$ All p are q.

Example 2.4: how all of this works

Suppose that $\mathbf{P} = \{n, p, q\}$. In this case, the language $\mathcal{A}$ would have exactly nine sentences. We encourage you to write them all out before going on.
Let $M = \{1, 2, 3, 4, 5\}$. Let $[\![n]\!] = \emptyset$, $[\![p]\!] = \{1, 3, 4\}$, and $[\![q]\!] = \{1, 3\}$. This is all we need to specify a model. We'll call this model $\mathcal{M}$.
The following sentences are true in $\mathcal{M}$: All n are n, All n are p, All n are q, All p are p, All q are p, and All q are q. The other three sentences in $\mathcal{A}$ are false in $\mathcal{M}$.

The first two of our example sentences are true in $\mathcal{M}$. For this, we use the fact that *the empty set is a subset of every set*[2].)

As soon as we have a formal semantics for a logical system, we get two further notions. These are used with every logic in this book, and we see them first in Definition 2.5 below.

Throughout this book, we use Greek letters like φ (phi), ψ (psi), and χ (chi) to denote sentences of whatever language is under discussion. We then will use Γ (the upper-case Greek letter gamma) to denote a *set* of sentences in whatever logical system we happen to be discussing. So in this chapter, Γ denotes a set of sentences in $\mathcal{A}$. We sometimes call Γ a set of *assumptions* or a *theory*, following the usage in modern logic.

[2]This means that our semantics commits us to a view of universal quantification that does not build in existential import. If one wishes to go the other way (in larger logical systems), then it is possible to do so. Either alternative leads to a well-behaved proof system.

Definition 2.5: semantic consequence

Let Γ be a set of sentences in the system, and $\mathcal{M}$ a model.
We say that $\mathcal{M} \models \Gamma$ iff $\mathcal{M} \models \varphi$ for every $\varphi \in \Gamma$.
We say that $\Gamma \models \varphi$ iff for all $\mathcal{M}$: if $\mathcal{M} \models \Gamma$, then also $\mathcal{M} \models \varphi$.
We read $\Gamma \models \varphi$ in a number of ways:

Every model of Γ is a model of φ
Γ *logically implies* φ
Γ *semantically implies* φ
φ is a *semantic consequence* of Γ.

We write $\Gamma \not\models \varphi$ if it is not the case that $\Gamma \models \varphi$. In plain terms, for *some* model $\mathcal{M}$, $\mathcal{M} \models \Gamma$ but $\mathcal{M} \not\models \varphi$.

Example 2.6: a semantic consequence

$$\{\textsf{All } p \textsf{ are } q, \textsf{All } n \textsf{ are } p\} \models \textsf{All } n \textsf{ are } q.$$

To justify what we say in Example 2.6 just above, we give a straightforward mathematical proof.[3] Let $\mathcal{M}$ be any model for $\mathcal{A}$, assuming that the underlying set $\mathbf{P}$ contains n, p, and q. Assume that $\mathcal{M}$ satisfies All p are q and All n are p. We must prove that $\mathcal{M}$ also satisfies All n are q. From our first assumption, $[\![p]\!] \subseteq [\![q]\!]$. From our second, $[\![n]\!] \subseteq [\![p]\!]$. It is a general fact about sets that the inclusion relation (written as $\subseteq$ here) is *transitive*, and so we conclude that $[\![n]\!] \subseteq [\![q]\!]$. This verifies that indeed $\mathcal{M}$ satisfies All n are q. And since $\mathcal{M}$ was arbitrary, we are done.

Incidentally, when Γ is a finite set that is written out explicitly on the left side of a $\models$ or $\vdash$ symbol, we usually omit the set braces $\{$ and $\}$ from the notation. So instead of writing what we saw in Example 2.6, we would usually write

$$\textsf{All } p \textsf{ are } q, \textsf{All } n \textsf{ are } p \models \textsf{All } n \textsf{ are } q.$$

Example 2.7: a failure of semantic consequence

$$\textsf{All } p \textsf{ are } q \not\models \textsf{All } q \textsf{ are } p.$$

To show that a given set Γ does not logically entail another sentence φ, we need to build a model $\mathcal{M}$ of Γ which is not a model of φ. In this example, Γ is $\{\textsf{All } p \textsf{ are } q\}$, and φ is All q are p. We can get a model $\mathcal{M}$ that does the trick by setting $M = \{1, 2\}$, $[\![p]\!] = \{1\}$, and $[\![q]\!] = \{1, 2\}$. For that matter, we could also use a different model, say $\mathcal{N}$, defined by $N = \{61\}$, $[\![p]\!] = \emptyset$, and $[\![q]\!] = \{61\}$.

[3] There is no real need to use a formal system to do mathematical proofs! They are much more readable when done informally.

Example 2.8: Challenge

Here is a more complicated example. Let

$$\Gamma = \left\{ \begin{array}{lll} \textsf{All } j \textsf{ are } k, & \textsf{All } j \textsf{ are } l, & \textsf{All } k \textsf{ are } l, \\ \textsf{All } l \textsf{ are } k, & \textsf{All } l \textsf{ are } m, & \textsf{All } k \textsf{ are } n, \\ \textsf{All } m \textsf{ are } q, & \textsf{All } p \textsf{ are } q, & \textsf{All } q \textsf{ are } p \end{array} \right\}$$

True or false: $\Gamma \models$ All p are n? If true, give a reason. If false, give a model of Γ where $[\![p]\!] \not\subseteq [\![n]\!]$.
The point here not to simply solve this particular problem, but to give an *algorithm* which would solve all problems of this type, and to prove your answer. We'll do this in Section 2.5.

2.2 Proof system

Up until now, we have dealt with *semantic concepts*, concepts defined in terms of models. At this point, move to the proof system. But before we do this, we want to say a few words about syntax and semantics.

Because the discussion is controversial, and potentially tricky, we mark it off with a "dangerous bend" sign. This device is originally due to Bourbaki. The basic difference is that semantic concepts are supposed to relate a formal language to abstract entities that stand in for the real world, and they are supposed to be models of human understanding, while syntactic concepts may be understood in a purely formal way, without any reference to meaning.

Definition 2.9 presents a *proof system*[4] for the language $\mathcal{A}$. The name of the proof system is A. (This is the same letter as the language, but in a different font.)

This proof system A has two *inference rules*. Each rule allows conclusions to be inferred from some set of premises. The conclusion of a rule is the sentence below the horizontal line, and the premise(s) are above the line.

Let us discuss the second rule, (BARBARA) first. The rule says that All p are q may be inferred from the two premises All p are n and All n are q. That is, a node in a tree may be labeled All p are q provided it has two nodes above it in the tree, one labeled All p are n for some atom n, and the other labeled All n are q.

The first rule, named (AXIOM), says that All p are p may label a node in a proof tree provided that node have *no nodes above it*. So no premises are required, and none may appear.

The name (BARBARA) comes from medieval logic. The reason for this is that logicians used A for universal assertions. (Of course, they did not use English, so it is purely a coincidence that A stands for *All*.) The rule has three As. Classical logicians interpolated consonants "randomly" to get the names of their rules, hence bArbArA. Our rule

[4]We also call this a *logical system*.

Logic 2.9: the proof system A for $\mathcal{A}$

$$\frac{}{\textsf{All } p \textsf{ are } p}\ \textsc{axiom} \qquad\qquad \frac{\textsf{All } p \textsf{ are } n \quad \textsf{All } n \textsf{ are } q}{\textsf{All } p \textsf{ are } q}\ \textsc{barbara}$$

(AXIOM) is so-named because an axiom in a proof system is like a rule of inference with no premises.

Definition 2.10: proof tree

Let Γ be a set of sentences in $\mathcal{A}$. A *proof tree over* Γ is a finite tree[a] $\mathcal{T}$ whose nodes are labeled with sentences, and each node is either a leaf node labeled with an element of Γ, or else matches one of the rules in the proof system A in Definition 2.9.

$\Gamma \vdash \varphi$ means that there is a proof tree $\mathcal{T}$ for over Γ whose root is labeled φ. We read this as Γ *proves* φ, or Γ *derives* φ, or that φ *follows from* Γ *in our proof system* A.

[a] See page 15 for more on trees.

Example 2.11: an example of a proof tree

Let Γ be

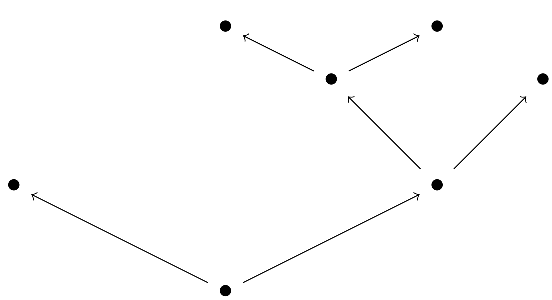
$$\{\textsf{All } l \textsf{ are } m, \textsf{All } q \textsf{ are } l, \textsf{All } m \textsf{ are } p, \textsf{All } n \textsf{ are } p, \textsf{All } l \textsf{ are } q\}$$

Let φ be $\textsf{All } q \textsf{ are } p$. We show in (2.2) below a proof tree showing that $\Gamma \vdash \varphi$.

We take

$$\frac{\textsf{All } q \textsf{ are } l \qquad \frac{\frac{\textsf{All } l \textsf{ are } m \quad \frac{}{\textsf{All } m \textsf{ are } m}\ \textsc{axiom}}{\textsf{All } l \textsf{ are } m}\ \textsc{barbara} \qquad \textsf{All } m \textsf{ are } p}{\textsf{All } l \textsf{ are } p}\ \textsc{barbara}}{\textsf{All } q \textsf{ are } p}\ \textsc{barbara} \tag{2.1}$$

This might not look like a tree with a labeling function, so we should explain this in more detail. The tree is



(2.2)

and then we label the various points with sentences in $\mathcal{A}$ that match what we showed in (2.1). Rather than give the function in any explicit way, we merely illustrate the way it works.

Again, the proof tree (technically speaking) is a tree together with a labeling function taking the nodes of the tree to sentences. But we never need to be so explicit, and it will be much better to think of the tree as it appears in (2.1). Similarly, whenever we refer to the *root* and *leaves* of the tree, we really mean the sentences that label the root and leaves.

Let us check that what we have in (2.1) really is a proof tree according to our definition. All of the leaves belong to Γ except for one: that is All m are m. This last leaf matches the first rule in Definition 2.9. That is, this rule allows us to use All m are m with no parents. All of the other nodes in the tree match the second rule. They all have two parents, and they are special cases of the rule (BARBARA).

Derivation trees come with the information of which rule is used at the various nodes of the tree, the way we do it in (2.1). But frequently we drop all the rules. So our derivation tree would look like

$$\frac{\text{All } q \text{ are } l \qquad \dfrac{\dfrac{\text{All } l \text{ are } m \quad \text{All } m \text{ are } m}{\text{All } l \text{ are } m} \qquad \text{All } m \text{ are } p}{\text{All } l \text{ are } p}}{\text{All } q \text{ are } p} \tag{2.3}$$

Finally, note also that some sentences from Γ are not used as leaves. This is permitted according to our definition. Also, there is a smaller proof tree that also shows that $\Gamma \vdash \varphi$: we could drop All m are m (The reason why have the rule (AXIOM) is so that that we can have one-element trees labeled with sentences of the form All l are l.)

We now have a proof system A for the language $\mathcal{A}$ of this chapter. As our notations indicate, the language does not automatically come with the proof system. The proof system is an "extra." For that matter, there could well be two different proof systems for the same language. As an artificial example, we could take our system A and add a rule that is redundant. For example, we could add

$$\frac{\text{All } w \text{ are } x \quad \text{All } x \text{ are } y \quad \text{All } y \text{ are } z}{\text{All } w \text{ are } z}$$

Then we would have a *different proof system*.

Soundness and completeness Earlier, we defined the semantics of $\mathcal{A}$, using models. From the models, we get the derived semantic notion $\Gamma \models \varphi$ as in Definition 2.5. The main technical work is to connect the semantic notion "$\Gamma \models \varphi$" with the proof-theoretic notion "$\Gamma \vdash \varphi$." This connection happens in two directions. The main definitions pertaining to this connection are found in Definition 2.12. Like much of the work in this chapter, these definitions will be used in all of the logical system we shall see.

Suppose we are given a logical language and a semantics for it, defining a notion $\Gamma \models \varphi$.

Suppose we also have a proof system for the same language, defining a notion $\Gamma \vdash \varphi$.

Definition 2.12: soundness and completeness

A proof system is *sound for a semantics* if whenever $\Gamma \vdash \varphi$, we also have $\Gamma \models \varphi$. A proof system is *complete for a semantics* if whenever $\Gamma \models \varphi$, we also have $\Gamma \vdash \varphi$.

Our next goal is to prove the soundness of the proof system with respect to the semantics. This proof uses *induction*, and so we turn to that topic. To prove something about all proof trees over a set Γ, we use *induction on proof trees over* Γ[5]. The statement that we need may be found in Definition 2.14.

We'll see examples of induction on proof trees in Proposition 2.13 and in Exercises 2.10 and 2.12.

Proposition 2.13: Soundness

If $\Gamma \vdash \varphi$, then $\Gamma \models \varphi$.

Proof. We prove this by induction on proof trees over Γ as stated in Definition 2.14. We take $S(\mathcal{T})$ to be the assertion:

$$\text{if the root of } \mathcal{T} \text{ is labeled } \varphi, \text{ then every } \mathcal{M} \models \Gamma \text{ also satisfies } \varphi \tag{2.4}$$

First, we show that $S(\mathcal{T})$ when $\mathcal{T}$ is a one-point tree labeled with a sentence from Γ. In this case, φ belongs to Γ. And so every model $\mathcal{M}$ of *all* sentences in Γ is a fortiori a model of φ.

Second, we show that $S(\mathcal{T})$ when $\mathcal{T}$ is proof tree over Γ whose root is justified by (AXIOM). In this case, the root of $\mathcal{T}$ is a sentence of the form $\mathsf{All}\ p\ \mathsf{are}\ p$. (Also, in this case, the entire tree is the root. But this is irrelevant.) Every sentence $\mathsf{All}\ p\ \mathsf{are}\ p$ is true in all models $\mathcal{M}$ whatsoever, regardless of whether $\mathcal{M}$ satisfies Γ or not. So $S(\mathcal{T})$ holds in this case.

Now assume that the root of $\mathcal{T}$ is $\mathsf{All}\ n\ \mathsf{are}\ q$, and right above this we have two trees whose roots are $\mathsf{All}\ n\ \mathsf{are}\ p$ and $\mathsf{All}\ p\ \mathsf{are}\ q$, respectively. Assume that every model of Γ satisfies $\mathsf{All}\ n\ \mathsf{are}\ p$, and also that every model of Γ satisfies $\mathsf{All}\ p\ \mathsf{are}\ q$. Fix a model $\mathcal{M}$ of Γ. Then in $\mathcal{M}$, $[\![n]\!] \subseteq [\![p]\!]$, and also $[\![p]\!] \subseteq [\![q]\!]$. Continuing to think about this fixed model, we also have $[\![n]\!] \subseteq [\![q]\!]$. But $\mathcal{M}$ was an arbitrary model of Γ. So we have shown than every model of Γ satisfies the root of $\mathcal{T}$. That is, $S(\mathcal{T})$ holds.

The paragraphs above show that for all proof trees $\mathcal{T}$ over Γ, every model of Γ satisfies the root of $\mathcal{T}$. Put another way, if $\Gamma \vdash \varphi$, then $\Gamma \models \varphi$. □

[5]If you don't know about induction, you might wish to consult some other sources to learn about induction in other contexts, especially *induction on numbers*. Proofs by induction will only play a minor role in what we do, so if you don't mind taking a few results on faith, it would be possible to continue reading, forgetting induction completely. However, the settings that we have are simple enough that you might even be able to pick up induction from the rest of the book.

Definition 2.14: the induction principle for proof trees over Γ

Let $S(\mathcal{T})$ be an assertion about proof trees. Suppose that

1. $S(\mathcal{T})$ whenever $\mathcal{T}$ is a one-point tree labeled with a sentence from Γ.
2. $S(\mathcal{T})$ whenever $\mathcal{T}$ is a proof tree over Γ whose root is justified by (AXIOM).
3. Let $\mathcal{T}$ be a proof tree over Γ whose root is justified by (BARBARA), and let $\mathcal{T}_1$ and $\mathcal{T}_2$ be the subtrees right above the root. If $S(\mathcal{T}_1)$ and also $S(\mathcal{T}_2)$, then $S(\mathcal{T})$.

Assuming these conditions, we have $S(\mathcal{T})$ for all proof trees $\mathcal{T}$ over Γ.

Remark 2.15: on soundness proofs

The soundness proofs of all the logical systems in this book are all pretty much the same as the one in Proposition 2.13. They are always inductions, and the crux of the matter is usually a simple fact about sets. (In Proposition 2.13, the crux of the matter is that the inclusion relation $\subseteq$ on subsets of a given set is always a transitive relation.) We almost never present the soundness proof in any detail.

Proposition 2.13 tells us that the formal logical system for $\mathcal{A}$ is not going to give us any bad results. (This is what *soundness* means.) Now this is a fairly weak point. If we dropped some of the rules, it would still hold. Even if we decided to be conservative and say that $\Gamma \vdash \varphi$ *never* holds, Proposition 2.13 would still hold. So the more interesting question to ask is whether the logical system is *strong* enough to prove everything it *should prove*. We want to know if $\Gamma \models \varphi$ implies that $\Gamma \vdash \varphi$. If this implication does hold for all Γ and φ, then we say that our system is *complete*.

2.3 Preliminaries: graphs and preorders

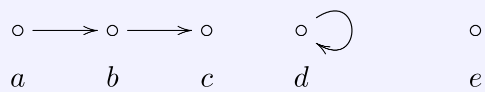
At this point, we have presented a language called $\mathcal{A}$, a semantics for it, and a sound proof system to go along with the semantics. The main theoretical work is to show that the system is complete: if $\Gamma \models \varphi$, then $\Gamma \vdash \varphi$. We also discuss the algorithmic properties of the system. That is, if someone gives us a (finite) set Γ and a sentence φ, how can we determine whether or not $\Gamma \vdash \varphi$? We shall study these soon, but first we need some preliminary notions from general mathematics.

Definition 2.16: graph

A *graph* is a pair $\mathcal{G} = (G, \to)$, where G is a set and $\to$ is a relation on G. The elements of G are called *nodes*, *vertices*, or *points*.

If g and h are nodes of $\mathcal{G}$, we usually write $g \to h$ to mean that g and h are related by the relation $\to$. But sometimes the style of exposition dictates other notation, such as $(g, h) \in \to$. Alternately, we might say that g is the *parent* of h. The relation is also called the *edge relation* of the graph. There is no requirement on this relation: G might have *loops* (i.e., we might have $g \to g$ for some, or even all nodes of G). But a graph need not have any loops at all. (For that matter, a graph need not have *any* nodes at all! The empty graph is perfectly fine for us. But as with all similar situations, other authors may differ on this point of usage.)

Example 2.17: a graph



Technically, we have drawn a picture of a graph $\mathcal{G}$, where the node set G is $\{a, b, c, d, e\}$, and the edge relation $\to$ is $\{(a, b), (b, c), (d, d)\}$.

Definition 2.18: the all-graph of a set

For each set Γ of sentences of $\mathcal{A}$ and each set G of atoms, we get a graph $\mathcal{G}_{\Gamma,G}$. The nodes of $\mathcal{G}_{\Gamma,G}$ are the elements of the given set G, and the relation $\to$ is defined by

$$x \xrightarrow[\Gamma]{} y \quad \text{iff} \quad \text{the sentence } \mathsf{All}\ x\ \mathsf{are}\ y \text{ belongs to } \Gamma$$

We call this the *all-graph of* Γ *on* G. Graphs of this form are the main reason why we introduce graphs in this section.

There are two special cases of this construction which interest us. One is where $G = \mathbf{P}$, the set of all atoms. The second is where we have a set Γ and another sentence φ, and G is the set of all atoms which occur in Γ or in φ.

Definition 2.19: $g \xrightarrow{*} h$ in a graph

If $\mathcal{G}$ is any graph, we write $g \xrightarrow{*} h$ to mean that there is a finite *path* from g to h following the edge relation in the graph. (The path might consist of 0 edges; in that case h would be the same as g. So we always have $g \xrightarrow{*} g$.)
The relation $\xrightarrow{*}$ is called the *reflexive-transitive closure* of the graph relation $\to$.

For example, in the graph $\mathcal{G}$ in Example 2.17, we have $a \xrightarrow{*} c$, using the path $a \to b \to c$. But paths can be of length 1 or even 0. The full listing of $\xrightarrow{*}$ is

$$a \xrightarrow{*} b \quad b \xrightarrow{*} c \quad a \xrightarrow{*} c \quad a \xrightarrow{*} a \quad b \xrightarrow{*} b \quad c \xrightarrow{*} c \quad d \xrightarrow{*} d \quad e \xrightarrow{*} e$$

Trees We defined proof trees for our logic in Section 2.2. But we neglected at that point to say what *trees* actually are.

Definition 2.20: tree

A *tree* is a graph with a designated node called the *root* with the property that every node is reachable from the root by a unique path. A *leaf* of a tree is a node with no successors (no outgoing edges).

Definition 2.21: preorder

A *preorder* is a pair $\mathbb{P} = (P \leq)$, where P is a set, and $\leq$ is a relation on P which is both *reflexive* and *transitive*.

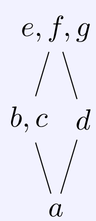
Reflexivity means that $p \leq p$ for all $p \in P$. Transitivity means that if $p \leq q$ and $q \leq r$, then also $p \leq r$.

Example 2.22

Perhaps the most natural example of a preorder would be the *power set preorder* on a set X. This is $(\mathcal{P}(X), \subseteq)$, where $\mathcal{P}(X)$ is the set of subsets of X, and $\subseteq$ is the *inclusion relation* on $\mathcal{P}(X)$: $A \subseteq B$ means that every element of A is also an element of B.

Here is another example of a preorder, a more artificial example, but one which allows us to introduce a points of notation and terminology. We take P to be the set $\{a, b, \ldots, g\}$. Then we take the poset to be as shown below:

A picture is worth a thousand words



$$\begin{array}{l} a \leq a, a \leq b, a \leq c, a \leq d, a \leq e, a \leq f, \\ a \leq g, b \leq b, b \leq c, b \leq e, b \leq e, b \leq f, \\ b \leq g, c \leq b, c \leq c, c \leq e, c \leq e, c \leq f, \\ c \leq g, d \leq d, d \leq e, d \leq f, d \leq g, e \leq e, e \leq f, \\ e \leq g, f \leq e, f \leq f, f \leq g, g \leq e, g \leq f, g \leq g \end{array}$$

On the left we have a picture, and on the right the same picture written out as a big set of pairs. You can see why most people prefer the picture.

Either way, we have a preorder $\mathbb{P} = (P, \leq)$. Note that $b \leq c \leq b$. We indicate this by writing $b \equiv c$, and we say that b and c are *equivalent* in the preorder. Notice that *equivalent* does not imply *identical* in a preorder. The picture of a preorder is called its *Hasse diagram*. As the lines above show, a picture is worth several lines of explicit information. We draw the preorder by putting a at the bottom, since it is $\leq x$ for all $x \in P$. We put b and c together, since $b \equiv c$, and since there is nothing between them and a. We do the same for d. There is no significance to the fact that b, c is left of d. Notice that the reflexivity condition is implicit in the picture. Similarly, the transitivity is also implicit. So we would never draw a separate line from a to e, for example.

There is a natural way of turning every graph $\mathcal{G}$ into a preorder $\mathbb{P}$.

Proposition 2.23

For any graph $(\mathcal{G}, \to)$, the structure $(\mathcal{G}, \xrightarrow{*})$ is a preorder.

Proof. The fact that we allow paths to have length 0 tells us that for each point g in $\mathcal{G}$, we have $g \xrightarrow{*} g$. Thus, $\xrightarrow{*}$ is reflexive. For the transitivity, suppose that $g \xrightarrow{*} h$ and $h \xrightarrow{*} i$. Then there is a path from g and h, and another path from h to i. Chaining the paths gives a path from g to i, showing that $g \xrightarrow{*} i$. □

We need one more general definition concerning preorders.

Definition 2.24: down-set

Let $\mathbb{P}$ be a preorder, and let $p \in P$. Then we define the *down-set* of p, written $\downarrow p$, to be

$$\downarrow p \quad = \quad \{q \in P : q \leq p\}.$$

We read $\downarrow p$ as "the down-set of p." The reason for the name is that in the Hasse diagram of the preorder, the down-set of an element is the set of points below (or equivalent to) that element.

For example, in the preorder shown on page 16,

$$\begin{array}{lcl} \downarrow a & = & \{a\} \\ \downarrow b & = & \{a, b, c\} \\ \downarrow e & = & \{a, b, c, e, f, g\} \end{array}$$

2.4 Completeness

The last section was a digression from our main thread in this chapter, the logical system A for $\mathcal{A}$. We now return to that thread and prove the completeness of A.

Definition 2.25: $u \leq_\Gamma v$

Let Γ be a set of sentences in $\mathcal{A}$, or in any logical language containing $\mathcal{A}$. Define $u \leq_\Gamma v$ to mean that

$$\Gamma \vdash \mathsf{All}\ u\ \mathsf{are}\ v.$$

As always, we simplify the notation by dropping the subscript Γ if it is clear from the context.

Proposition 2.26

For all sets Γ, $(\mathbf{P}, \leq_\Gamma)$ is a preorder.

Proof. To check that $\leq$ is reflexive, let $u \in \mathbf{P}$. Then we have a one-line proof of All u are u. Indeed, this is the point of having this kind of no-premise rule in the proof system. For the transitivity, we put together a proof of $u \leq v$ with a proof of $v \leq w$ to show that $u \leq w$. □

We are going to the fact that $\leq$ is a preorder frequently in this book, and often without explicitly mentioning it.

We now have an important model construction. Starting with a set $\mathbf{P}$ of atoms and a theory Γ in our language $\mathcal{A}$, we build a model $\mathcal{M}$ *from the syntax and the proof theory.*

Definition 2.27: the canonical model of a set Γ

$$\begin{array}{rcl} M & = & \mathbf{P} \\ [\![u]\!] & = & \downarrow u \end{array}$$

It is worthwhile spelling out $[\![u]\!]$ in more detail, using the definitions of the preorder $\leq$ and of down-sets in general from Definition 2.24:

$$\begin{array}{rcl} M & = & \mathbf{P} \\ [\![u]\!] & = & \{v : \Gamma \vdash \textsf{All } v \textsf{ are } u\} \end{array}$$

Here is a verbal description of the model: we are taking the set of atoms underlying the language $\mathcal{A}$ to be the universe of our model, and we interpret a given atom u as the set of all atoms v which we can prove to be included in u, using Γ as a set of assumptions. $\mathcal{M}$ is called the *canonical model of* Γ.

Lemma 2.28

The canonical model of Γ satisfies Γ.

Proof. Suppose All p are q belongs to Γ. Then $p \leq q$. We must show that $[\![p]\!] \subseteq [\![q]\!]$. Let $v \in [\![p]\!]$. Then by Definition 2.27, $v \leq p$. Since $\leq$ is transitive, $v \leq q$. That is, $v \in [\![q]\!]$. This for all v shows our result. □

Theorem 2.29: Completeness

The logic A of Definition 2.9 is complete for $\mathcal{A}$: If $\Gamma \models$ All p are q, then $\Gamma \vdash$ All p are q.

Proof. Suppose that $\Gamma \models$ All p are q. We must prove that $\Gamma \vdash$ All p are q. Consider $\mathcal{M}$ from Definition 2.27. By Lemma 2.28, $\mathcal{M}$ satisfies all sentences in Γ. By hypothesis, we see that All p are q is true in $\mathcal{M}$. Thus $[\![p]\!] \subseteq [\![q]\!]$. We always have $p \leq p$, and so $p \in [\![p]\!]$. Hence $p \in [\![q]\!]$. This means that $p \leq q$, so that $\Gamma \vdash$ All p are q, as desired. □

Remark 2.30

The proof in fact shows that if $\Gamma \not\models$ All p are q, then All p are q is false in $\mathcal{M}$. So we have a very special and important fact: to see whether a given sentence φ follows from Γ or not, we only have to see whether φ is true or false in one model, $\mathcal{M}$. We say that this model $\mathcal{M}$ is a *characteristic model of* Γ.

2.5 Algorithmic analysis

The original definition of the entailment relation $\Gamma \models \varphi$ involves looking at *all* models of the language. If we are given Γ and φ and we want to know whether or not $\Gamma \models \varphi$, we can say "no" by producing a *counter-model*: a model of Γ where φ fails. If we want to say "yes", the easiest way would be to provide a derivation in our proof system of φ from Γ. This would show that $\Gamma \vdash \varphi$, and then by soundness (Proposition 2.13), we would know that indeed $\Gamma \models \varphi$.

But suppose that we are given Γ and φ, and we don't know whether or not $\Gamma \models \varphi$. For example, suppose we are faced with Γ and φ from Example 2.8. What happens now? We could try building a counter-model and searching for a proof at the same time. This would work, but we would like something better. That "something better" would be an algorithm that was more detailed and more organized than a blind search, and that either gave a derivation or a counter-model. This is the topic of this section. The centerpiece of this section is Theorem 2.31 just below, a *refined version* of the completeness argument which we saw in Section 2.4. Theorem 2.31 leads to Algorithm 2.33.

Theorem 2.31

Let Γ be any set of sentences in $\mathcal{A}$, let G be any set of atoms which includes the atoms in Γ, and let $\mathcal{G}$ be the all-graph of Γ on G. Let $\xrightarrow[\Gamma]{*}$ be the reflexive-transitive closure of the graph relation in $\mathcal{G}$. Let $p, q \in G$. Then the following are equivalent:

1. $p \xrightarrow[\Gamma]{*} q$.
2. $\Gamma \vdash$ All p are q.
3. $\Gamma \models$ All p are q.

Proof. (1)⇒(2): we show by induction on the number n that if q is reachable from p in $\mathcal{G}$ by a path of length n, then $\Gamma \vdash$ All p are q. If $n = 0$, then $p = q$, and we have a one-point proof tree of All p are p. Suppose our result is true for n, and also that q is reachable from p in $\mathcal{G}$ by a path of length $n + 1$. Fix such a path. Suppose that q' is the point on the path right before q. Then q' is reachable from p in $\mathcal{G}$ by a path of length n. So by induction hypothesis, we have $\Gamma \vdash$ All p are q'. Since there is an edge from q' to q, we know that the sentence All q' are q belongs to Γ. So we can take $\mathcal{T}$ and add a step at the end:

$$\dfrac{\begin{matrix}\vdots & \\ \textsf{All } p \textsf{ are } q' & \textsf{All } q' \textsf{ are } q\end{matrix}}{\textsf{All } p \textsf{ are } q}\ \textsc{barbara}$$

This is a proof tree over Γ, and it shows that $\Gamma \vdash$ All p are q.

(2)⇒(3): this is the soundness result for the logic which we saw in Proposition 2.13.

(3)⇒(1): we show the contrapositive. Assume that $p \not\xrightarrow[\Gamma]{*} q$. Let $\mathcal{N}$ be the following model: the universe is the set G, and the interpretation is given by

$$[\![u]\!] \quad = \quad \{v \in G : v \xrightarrow[\Gamma]{*} u\} \tag{2.5}$$

We claim that $\mathcal{N} \models \Gamma$. To see this, suppose that Γ contains the sentence All x are y. We show that $[\![x]\!] \subseteq [\![y]\!]$. Let $z \in [\![x]\!]$. That is, $z \in G$ and there is a path from z to x in $\mathcal{G}_{\Gamma,G}$. The atom y occurs in Γ and hence belongs to G. Then taking the path from above and adding the edge from x to y shows that $z \xrightarrow[\Gamma]{*} y$. This for all x shows that indeed $[\![x]\!] \subseteq [\![y]\!]$; and the preceding observation for all sentences in Γ shows that $\mathcal{N} \models \Gamma$. By definition of $\xrightarrow[\Gamma]{*}$, $p \xrightarrow[\Gamma]{*} p$. Also, $p \in G$ by hypothesis. So $p \in [\![p]\!]$. Our overall assumption that $p \not\xrightarrow[\Gamma]{*} q$ tells us that $p \notin [\![q]\!]$. Hence $\mathcal{N} \not\models$ All p are q.

This completes the proof. □

Corollary 2.32

Let $\Gamma \cup \{$All p are $q\}$ be a set of sentences in $\mathcal{A}$. Let G be the set of atoms in Γ together with p and q. Let $\mathcal{G}$ be the all-graph of Γ on G. Then the following are equivalent:

1. $\Gamma \models \varphi$.

2. $p \xrightarrow[\Gamma]{*} q$ in $\mathcal{G}$.

Corollary 2.32 leads to the algorithm presented in Algorithm 2.33. More to the point, Corollary 2.32 shows that the algorithm is correct. Given a finite set Γ and a sentence φ (say, All p are q), we first take the set of atoms in Γ. We add p and q to this set if they are not there already, and then the resulting set is called G. We construct the all-graph of Γ on G. The all-graph is directly read from Γ. There is a well-known algorithm which, given any relation R finds the reflexive-transitive closure R^*. So we apply this algorithm to the all-graph of Γ on G, and we ask whether or not $p \xrightarrow[\Gamma]{*} q$. If so, then the proof of (1) ⇒ (2) in Theorem 2.31 shows how to turn the path into a derivation in the logic. If not, then the model $\mathcal{N}$ constructed in (3) ⇒ (1) is a *counter-model*: a model of Γ where the putative conclusion All p are q is false.

Algorithm 2.33: given a finite set $\Gamma \subseteq \mathcal{A}$ and a sentence $\varphi = \textsf{All } p \textsf{ are } q$, tell whether or not $\Gamma \vdash \varphi$

let G be the set of atoms in $\Gamma \cup \{\varphi\}$.
let $\mathcal{G} = (G, \xrightarrow[\Gamma]{})$ be the all-graph of Γ on G. ▷ see Definition 2.18.
let $\xrightarrow[\Gamma]{*}$ be the reflexive-transitive closure of $\xrightarrow[\Gamma]{}$.
if $p \xrightarrow[\Gamma]{*} q$ **then**
    a path in $\mathcal{G}$ gives a proof tree in the logic, showing that $\Gamma \vdash \varphi$.
else ▷ we build a counter-model $\mathcal{N} = (N, [\![\ ]\!])$
    $N \leftarrow G$
    for all $u \in \mathbf{P}$ **do**
        $[\![u]\!] = \{v : v \xrightarrow[\Gamma]{*} u\}$.
    end for
end if

Example 2.34: a return to Example 2.8

We challenged you to see whether $\Gamma \vdash \textsf{All } p \textsf{ are } n$ or not. At this point, we can meet the challenge.

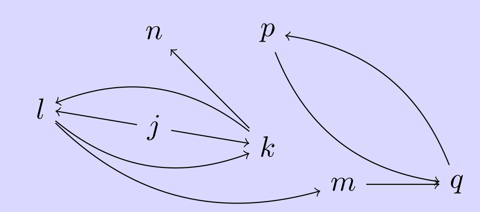
The relevant set G here is $\{j, k, l, m, n, p, q\}$. We show Γ below, for convenience, and its all-graph, the preorder associated to that graph, and finally the structure of the canonical model.

Notice that the preorder is displayed as a Hasse diagram. For example, since $j \le k$, we draw j below k. Note also that $k \le l$ and $l \le k$. We indicate this in the picture by situating k and l together.

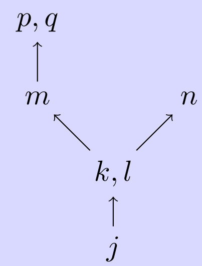
the set Γ from Example 2.8

All j are k,
All j are l,
All k are l,
All l are k,
All l are m,
All k are n,
All m are q,
All p are q,
All q are p

the all-graph



the associated preorder



our counter-model showing that $\Gamma \not\vdash$ All p are n

$$\begin{array}{lcl cl}
[\![j]\!] & = & \{j\} & & \\
[\![k]\!] & = & \{j,k,l\} & = & [\![l]\!] \\
[\![m]\!] & = & \{j,k,l,m\} & & \\
[\![n]\!] & = & \{j,k,l,n\} & & \\
[\![p]\!] & = & \{j,k,l,m,p,q\} & = & [\![q]\!]
\end{array}$$

At the end, we interpreted each atom u according to (2.5): the interpretation of each u is the set of atoms x such that there is a path from x to u in the all-graph. Equivalently, this is the set of x which are $\leq u$ in the preorder.

Comments There are a few comments to be made at this point. These are based on questions which I have received in teaching this material, or similar material, to bright students.

(1) *Doesn't this proof confuse syntax and semantics?* This is a good question, since we are building a model out of the "material from the syntax" (in this case, the atoms). But when one thinks about it, there is nothing wrong with building a model out of chairs, numbers, abstract objects, or even the same objects that we used in the syntax. It is more interesting that the interpretation function $[\![\ ]\!]$ in the model was defined in terms of a syntactic notion, the proof system. Again, we are free to define the semantics of a model any way we like. It is *interesting* that we prove completeness in this way. But in a sense it should not be such a surprise. For completeness is about a relation between syntax and semantics, and so it makes sense that it involves a single structure that has aspects of both.

(2) *I thought that the semantic assertion $\Gamma \models \varphi$ meant that all models of Γ are again models of φ. How is it that we only argue completeness on one particular model rather than many models?* It is true that our semantic assertion $\Gamma \models \varphi$ is a statement about *all* models. But this does not mean that in proving something we need to *use* all the models. In a sense, the question can be turned around to make an observation: the model $\mathcal{M}$ we built from a set Γ is as "bad" as any model could possibly be! It makes true any sentence which does not follow from Γ. So once we have built a single model that covers for all the models, we can use that one model to prove completeness.

What you need to know to go on At this point, you should understand a few things well: the *syntax* of the language in this chapter, beginning with the set $\mathbf{P}$ of atoms; the *semantics* of the language, including the definition of a *model* and the definition of when a model *satisfies* a given sentence; the *proof theory*, defined in terms of two *rules of inference*; the statements of *soundness* and *completeness*; the proof that our logical system is complete (it was done twice); the general issue of *algorithms for proof search* *in a given logic*; how the algorithm works for the particular logic in this chapter; the definition of a *counter-model* to an assertion of the form $\Gamma \vdash \varphi$.

The reason that you should understand all of this is because in most of the remaining chapters, we shall see an ever-increasing array of logics. Each time, we'll see all of the *iitalicized* points; of course, the specific details will vary with the logic. It would be good to really understand all of the points above. One way to be sure would be to write out explanations of everything mentioned in the last paragraph. Be sure to use your own words, but also to use all of the symbols that we introduced in this chapter.

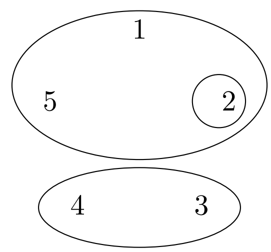
2.6 Exercises

Exercise 2.1. Suppose that our set $\mathbf{P}$ of nouns has the three words hawks, birds, and turtles. Here is a model which we'll call $\mathcal{M}$. The universe is $\{1, 2, 3, 4, 5\}$. The rest of the structure is shown below.

$[\![\text{hawks}]\!] = \{2\}$

$[\![\text{birds}]\!] = \{1, 2, 5\}$

$[\![\text{turtles}]\!] = \{3, 4\}$



Say whether each of the following sentences is true or false in this model $\mathcal{M}$.

1. All hawks are birds
2. All hawks are hawks
3. All hawks are turtles

Then find a single model (different from $\mathcal{M}$) where the sentence All turtles are birds is true and All hawks are birds is false.

Exercise 2.2. Here is an exercise on proof trees. Let

$$\Gamma = \{\text{All } x \text{ are } y, \text{All } y \text{ are } z, \text{All } z \text{ are } x\}.$$

Figure 2.1 shows six trees. Which are proof trees over Γ according to the strict letter of Definition 2.2?

Exercise 2.3. Here is an exercise for people getting started with proof trees and models. Let $\mathbf{P} = \{x, y, z, w\}$. Let

$$\Gamma = \{\text{All } x \text{ are } y, \text{All } x \text{ are } z, \text{All } w \text{ are } z\}.$$

$$\frac{}{\textsf{All } x \textsf{ are } x}\ \textsc{axiom} \qquad \textsf{All } y \textsf{ are } x \qquad \textsf{All } x \textsf{ are } y$$

$$\frac{\dfrac{}{\textsf{All } z \textsf{ are } z}\ \textsc{axiom} \quad \dfrac{}{\textsf{All } z \textsf{ are } z}\ \textsc{axiom}}{\textsf{All } z \textsf{ are } z}\ \textsc{barbara}$$

$$\frac{\textsf{All } z \textsf{ are } x \quad \textsf{All } x \textsf{ are } y}{\textsf{All } z \textsf{ are } y}\ \textsc{barbara}$$

$$\frac{\textsf{All } z \textsf{ are } y \quad \textsf{All } y \textsf{ are } z}{\textsf{All } z \textsf{ are } z}\ \textsc{barbara}$$

Figure 2.1: Trees from Exercise 2.2.

1. Since $\mathbf{P}$ has 4 elements, there are $4^2 = 16$ sentences that can be built from $\mathbf{P}$. List all 16.
2. Which of your 16 sentences φ in the last part are provable from Γ? For each such sentence φ, give a proof tree which has φ as the root, and with all leaves in Γ.
3. For each φ from part (1) which is *not* on your list in part (2), give a model of Γ where φ is false.

[This exercise would be long if done completely. One way to make it shorter would be to drop any of the sentences in Γ. Then the number of sentences in part (1) would be 9 instead of 16.]

Exercise 2.4. Here is an exercise just to see if you got the main points of this chapter. Let

$$\Gamma \quad = \quad \{\textsf{All } x \textsf{ are } y, \textsf{All } z \textsf{ are } y, \textsf{All } y \textsf{ are } w\}.$$

Let φ be All x are z. For each of the following assertions, say whether true or false, with a reason:

1. $\Gamma \models \varphi$.
2. $\Gamma \vdash \varphi$.
3. $x \leq_\Gamma w$.
4. $x \underset{\Gamma}{\rightarrow} w$.
5. $x \overset{*}{\underset{\Gamma}{\rightarrow}} w$.

If you are using the soundness or completeness of the logic A in any part, please be sure to note this. That is, please be aware when you are using significant facts!

Exercise 2.5. Suppose that $\Gamma \vdash \varphi$. For each of the following assertions, tell whether it is True or False, with a reason.

1. There are infinitely many proof trees over Γ which show that $\Gamma \vdash \varphi$.
2. Any two proof trees over Γ which show that $\Gamma \vdash \varphi$ must share a leaf. (The leaves of a proof tree are the sentences at the top which come from Γ.)
3. If a sentence of the form $\mathsf{All}\ p\ \mathsf{are}\ p$ occurs in a derivation tree, then that occurrence must be at a leaf.

Exercise 2.6. This exercise and the next ones develop a different way to prove the completeness of the logic A. Let Γ be the set in Exercise 2.4. As you saw in that exercise, $\Gamma \not\vdash \mathsf{All}\ y\ \mathsf{are}\ x$. Now we make a model $\mathcal{N}$ where $N = \{*\}$, some one-element set, and

$$\begin{array}{lllclll} [\![x]\!] & = & \emptyset & \qquad & [\![z]\!] & = & \emptyset \\ [\![y]\!] & = & \{*\} & & [\![w]\!] & = & \{*\} \end{array}$$

1. Does $\mathcal{N} \models \Gamma$? That is, are all sentences in Γ true in $\mathcal{N}$?
2. Does $\mathcal{N} \models \mathsf{All}\ y\ \mathsf{are}\ x$?
3. Check that for all variables $u \in \{x, y, z, w\}$,

$$[\![u]\!] \quad = \quad \begin{cases} \{*\} & \text{if } \Gamma \vdash \mathsf{All}\ y\ \mathsf{are}\ u \\ \emptyset & \text{otherwise} \end{cases} \tag{2.6}$$

   Note that u is a variable ranging over set of atoms $\{x, y, z, w\}$. But y on the right of (2.6) is just y, it's not a variable ranging over anything.

   To solve this problem plug the atoms x, y, z and w in for u in (2.6), and then check that each time, the two sides of the equation really are the same set.

   [You may use the results which you obtained in Exercise 2.4.]

Exercise 2.7. Here is a set Γ:

$$\begin{array}{lll} \mathsf{All}\ a\ \mathsf{are}\ b & \qquad & \mathsf{All}\ c\ \mathsf{are}\ d \\ \mathsf{All}\ a\ \mathsf{are}\ c & & \mathsf{All}\ a\ \mathsf{are}\ e \\ \mathsf{All}\ c\ \mathsf{are}\ e & & \end{array}$$

Then $\Gamma \not\models \mathsf{All}\ d\ \mathsf{are}\ b$. The point of this problem is to give *two* models of Γ where $\mathsf{All}\ d$ $\mathsf{are}\ b$ is false.

1. Find the canonical model of Γ, and check that $\mathsf{All}\ d\ \mathsf{are}\ b$ is false in that model.

2. Find a model $\mathcal{M}$ *with just one element* such that $\mathcal{M} \models \Gamma$ but $\mathcal{M} \not\models$ All d are b. [Hint: You can do this by modifying the model in Exercise 2.6. That is, you use a model $\mathcal{M}$ with $M = \{*\}$, and with the interpretation function given by something like (2.6). The only difference is that we don't want y on the right, we want .... You can also get a one-point model this by using Exercise 2.8 just below. On the other hand, some people might find Exercise 2.8 easier to think about after working on *this* exercise.]

Exercise 2.8. This exercise is a continuation of Exercises 2.6 and 2.7. Let Γ be any set, not necessarily the one in Exercise 2.4. Fix atoms p and q. Suppose that $\Gamma \models$ All p are q. Consider the model $\mathcal{N}$ where $N = \{*\}$, some one-element set, and

$$[\![u]\!] \quad = \quad \begin{cases} \{*\} & \text{if } \Gamma \vdash \textsf{All } p \textsf{ are } u \\ \emptyset & \text{otherwise} \end{cases} \tag{2.7}$$

Please note that in (2.7), the atom p is the same one as in the sentence All p are q that we are considering throughout this problem.

1. Show that $\mathcal{N} \models \Gamma$.
2. Show that $\mathcal{N} \models$ All p are q. [Hint: Use part (1).]
3. Use part (2) to show that $\Gamma \vdash$ All p are q.

Please note that this problem is more abstract and more general than Exercises 2.6 and 2.7. We don't know anything about Γ except that $\Gamma \models$ All p are q. So this is good: a more abstract and general result is a better one because it can be used in more settings.

Exercise 2.9. This exercise has to do with an aspect of the canonical model defined in Definition 2.27). Recall that we took M to be $\mathbf{P}$, the set of all atoms with underlies our language in this chapter. If we had taken M to be the set of atoms which occur in Γ, would our construction have worked? That is, what (if anything) would have to change in the rest of this section?

Exercise 2.10. Let Γ be a set of sentences in $\mathcal{A}$. Prove that for all p and q,

$$p \overset{*}{\underset{\Gamma}{\to}} q \quad \text{iff} \quad p \leq_\Gamma q$$

[Hint: you will need to use induction, in two forms. One direction of this exercise requires induction on numbers; the other uses induction on proofs in our proof system.]

Exercise 2.11. True or false: if a logical system is complete, one cannot delete any rules and still have a complete system. Be sure to give a reason for your answer.

Exercise 2.12. Suppose that Γ is the empty set of sentences. Note that $\emptyset \models \varphi$ means that φ is true in all models. (So what does $\emptyset \not\models \varphi$ mean?) Note also that $\emptyset \vdash \varphi$ means that there is a proof tree all of whose leaves are from the rule (AXIOM), so the leaves are all $\mathsf{All}\ x\ \mathsf{are}\ x$ for some variable x.

Your question: Under what conditions on p and q will we have $\emptyset \models \mathsf{All}\ p\ \mathsf{are}\ q$?

1. Give a semantic reason for your answer, one having to do with models.

2. Give a syntactic reason for your answer, one having to do with the proof system. In this part, you will need to use induction on proofs.

Exercise 2.13. Let $\mathbb{P} = (P, \leq)$ be any preorder, and let $\downarrow$ be the function from P to its power-set $\mathcal{P}(P)$ given by

$$\downarrow p \quad = \quad \{q \in P : q \leq p\}$$

Show that $\downarrow$ is a *monotone* function. That is, show that if $p \leq q$, then $\downarrow p \subseteq \downarrow q$. (Monotone functions in various settings will be important later in the book, and this is a first exposure.)

Exercise 2.14. Let Γ be a theory in any logical system extending A. Let $\mathcal{M}$ be any model with the property that

$$[\![\ ]\!] : \mathbf{P} \to \mathcal{P}(M)$$

is a monotone function (see Exercise 2.13). That is, we are assuming about $\mathcal{M}$ that if $p \leq_\Gamma q$, then $[\![p]\!] \subseteq [\![q]\!]$. Prove that $\mathcal{M} \models \Gamma$. [This is not a hard exercise, but it is one that will be used all the time.]

Exercise 2.15. Here is another logical system. Instead of atoms denoting subsets of a given set, our syntax starts with a set $\mathbf{N}$ of *names*. We use letters like a, b, etc. for names here. Then we take as sentences the expressions $a = b$, where $a, b \in \mathbf{N}$. For the semantics, we start with a set M and interpret a name a by an *element* $[\![a]\!] \in M$. (Again, we are not interpreting names using subsets the way we did for atoms in this chapter.) This gives us the definition of a *model* $\mathcal{M}$.

1. As soon as we have a semantics, we get notions like $\Gamma \models \varphi$ for this new language. (Note that for purposes of this problem, φ is a sentence like $a = b$.) State what $\Gamma \models \varphi$ means, and then give an example of a set Γ and a sentence φ where $\Gamma \models \varphi$ holds, and another example where it does not hold.

2. Find a logical system that defines a notion $\Gamma \vdash \varphi$, and give an example of a proof in your system. That is, what are the rules of your logic?

3. Prove that your system is complete.

Exercise 2.16. If you have worked Exercise 2.15, then the next step is to combine your system from that exercise with the logical system for $\mathcal{A}$. We start with sets $\mathbf{P}$ and $\mathbf{N}$ of atoms and names, respectively. In the syntax, we use sentences of the following forms:

$$\begin{array}{l} \textsf{All } p \textsf{ are } q \\ a = b \\ a \textsf{ isa } p \end{array}$$

The semantics is just what you would expect. In particular, in a given model $\mathcal{M}$, we say that a isa p is true if $[\![a]\!] \in [\![p]\!]$. The proof theory for this system should contain the rules in Definition 2.9, the rules in your system from Exercise 2.15, and the two extra rules below:

$$\frac{a \textsf{ isa } p \quad \textsf{All } p \textsf{ are } q}{a \textsf{ isa } q} \qquad \frac{a = b \quad a \textsf{ isa } p}{b \textsf{ isa } p}$$

Prove the completeness of the system.

Exercise 2.17. Suppose that $\mathbf{P}$ is the infinite set $\{x_1, x_2, \ldots,\}$, and let

$$\Gamma \quad = \quad \{\textsf{All } x_1 \textsf{ are } x_2, \textsf{All } x_2 \textsf{ are } x_3, \ldots, \textsf{All } x_n \textsf{ are } x_{n+1}, \ldots\}$$

Prove that if $\Gamma \vdash$ All x_i are x_j, then $i \leq j$. Use induction on proof trees over Γ.