Chapter 3

The Formal Syntax of Propositional Logic

Next, we will study the formal language of propositional logic.

By *syntax* we generally understand a set of rules for writing correctly. As mentioned previously, in this chapter we develop an artificial language, that we call *formal propositional logic* (or simply *propositional logic*). The formal syntax of propositional logic refers to the rules for writing correctly in this language.

3.1 Alphabets in Computer Science

A set is called an *alphabet* in computer science if we use the elements of the set to make up words. The elements of an alphabet are called *symbols*. Most of the time, an alphabet is a finite set (but not in the case of propositional logic).

What is the difference between an alphabet and a set? A priori, none. The intention matters – what we plan on doing with the elements of the set/alphabet. We use the elements of an alphabet to create *words*. A *word* over an alphabet is a sequence of symbols in the alphabet.

Example 11. *Consider the alphabet* $X = \{0, 1\}$. *The strings/sequences of symbols* `001010`, `101011` *and* `1` *are examples of words of the alphabet* X.

The empty word, formed of 0 symbols of the alphabet, is usually denoted by ϵ.

3.2 The Alphabet of Propositional Logic

The language of propositional logic is made of *propositional formulae*, which model propositions, as discussed in the previous chapter.

Propositional formulae are strings (sequences of characters) over the *alphabet of propositional logic*.

The alphabet of propositional logic is the union of the following sets:

1. $A = \{\mathtt{p}, \mathtt{q}, \mathtt{r}, \mathtt{p}', \mathtt{q}_1, \ldots\}$ is an infinite set of *propositional variables* that we fix from the very beginning;

2. $\{\neg, \wedge, \vee\}$ is the set of *logical connectives*;

3. $\{(,)\}$ is the set of auxiliary symbols; in our case, it consists of two symbols called *brakets*.

The set $L = A \cup \{\neg, \wedge, \vee, (,)\}$ is called the alphabet of propositional logic. Here are a few examples of words over L:

1. $))\mathtt{p} \vee \wedge$;
2. $\vee \vee \neg(\mathtt{p})$;
3. $\mathtt{p}$;
4. $\mathtt{ppp}$;
5. $\neg(\mathtt{p} \vee \mathtt{q})$.

Words, or strings, are simply sequences of symbols of the alphabet L. Some of these words will be formulae or, equivalently, well-formed formulae (wff). Some authors prefer to use the terminology *wff*, but we will simply use *formula* by default in these lecture notes.

As an example, the last word above, $\neg(\mathtt{p} \vee \mathtt{q})$ is a formula of propositional logic, but $\vee \vee \neg(\mathtt{p})$ is not. The following definition captures exactly the set of propositional formulae.

3.3 Propositional Formulae

Definition 12 (The Set of Propositional Formulae ($\mathbb{PL}$))**.** *The set of formulae of propositional logic, denoted $\mathbb{PL}$ from hereon, is the smallest set of words over L satisfying the following conditions:*

• *Base Case Any propositional variable, seen as a 1-symbol word, is in $\mathbb{PL}$ (equivalently, $A \subseteq \mathbb{PL}$);*

• *Inductive Step i* *If* $\varphi \in \mathbb{PL}$, *then* $\neg\varphi \in \mathbb{PL}$ *(equivalently, if the word* φ *is a propositional formula, then so is the word starting with the symbol* $\neg$ *and continuing with the symbols in* φ*);*

• *Inductive Step ii* *If* $\varphi_1, \varphi_2 \in \mathbb{PL}$, *then* $(\varphi_1 \wedge \varphi_2) \in \mathbb{PL}$ *(equivalently, exercise);*

• *Inductive Step iii* *If* $\varphi_1, \varphi_2 \in \mathbb{PL}$, *then* $(\varphi_1 \vee \varphi_2) \in \mathbb{PL}$ *(equivalently, exercise);*

Here are examples of elements of $\mathbb{PL}$:

$$\mathtt{p} \quad \mathtt{q} \quad \neg\mathtt{p} \quad \neg\mathtt{q} \quad \neg\mathtt{p}' \quad \neg\neg\mathtt{p}_1 \quad (\mathtt{p} \vee \mathtt{q}) \quad (\mathtt{p} \wedge \mathtt{q})$$

$$\neg(\mathtt{p} \vee \mathtt{q}) \quad (\neg\mathtt{p} \wedge \neg\mathtt{q}) \quad \neg(\neg\neg\mathtt{p} \vee \mathtt{p}) \quad ((\mathtt{p} \vee \mathtt{q}) \wedge \mathtt{r}) \quad (\mathtt{p} \vee (\mathtt{q} \wedge \mathtt{r}))$$

Here are examples of words not in $\mathbb{PL}$:

$$\mathtt{pp} \qquad \mathtt{q}\neg\mathtt{q} \qquad \mathtt{q} \wedge \neg\mathtt{p} \qquad \mathtt{p} + \mathtt{q}$$

The definition of the set $\mathbb{PL}$ is an example of an *inductive definition*. Such definitions are really important in computer science and it is a must to understand them very well. In inductive definitions of sets, there are usually some base cases, which say what *base* elements are part of the set and some inductive cases, which explain how to obtain new elements of the set from old elements. Another important part of an inductive definition is the minimality constraint, which says that nothing other than what is provable by the base case(s) and the inductive case(s) belongs to the set. It can be shown that the above set $\mathbb{PL}$ exists and is unique, but the proof is beyond the scope of this course.

3.4 Showing That a Word Is In $\mathbb{PL}$

We can show that a word belongs to $\mathbb{PL}$ by explaining how the rules of the inductive definition were applied. Here is an example of a proof that $\neg(\mathtt{p} \vee \mathtt{q}) \in \mathbb{PL}$:

1. $\mathtt{p} \in \mathbb{PL}$ (by the Base Case, because $\mathtt{p} \in A$);
2. $\mathtt{q} \in \mathbb{PL}$ (by the Base Case, as $\mathtt{q} \in A$);
3. $(\mathtt{p} \vee \mathtt{q}) \in \mathbb{PL}$ (by the Inductive Case iii, with $\varphi_1 = \mathtt{p}$ and $\varphi_2 = \mathtt{q}$);
4. $\neg(\mathtt{p} \vee \mathtt{q}) \in \mathbb{PL}$ (by the Inductive Case i, with $\varphi = (\mathtt{p} \vee \mathtt{q})$).

We can rearrange the above into an equivalent *annotated construction tree* for the formula $\neg(p \vee q)$:

$$\dfrac{\dfrac{\dfrac{}{p \in \mathbb{PL}}\ \textit{Base Case} \qquad \dfrac{}{q \in \mathbb{PL}}\ \textit{Base Case}}{(p \vee q) \in \mathbb{PL}}\ \textit{Inductive Case iii}}{\neg(p \vee q) \in \mathbb{PL}}\ \textit{Inductive Case i}$$

You will see this notation several times in Computer Science and it is worth getting to know it. Each line is called an inference; below each line is the conclusion of the inference and above the lines are the hypotheses (0, 1 or more). Besides each lines is the name of the rule that was applied.

If we drop all annotation, we obtain the following bare *construction tree* for the formula:

$$\dfrac{\dfrac{\overline{p} \qquad \overline{q}}{(p \vee q)}}{\neg(p \vee q)}$$

It is easy to see that a word belongs to $\mathbb{PL}$ iff there is a construction tree for it.

3.5 The Main Connective of a Formula

A formula that consists of a single propositional variable, such as p or q, is called an *atomic formula*. This explains why the set A of propositional variables is called A (A stands for *atomic*).

More complex formulae, such as $\neg p$ or $p \vee q$, are called *molecular* (to distinguish them from atomic formulae).

Each molecular formula has a *main connective*, which is given by the last inference in its construction tree. For example, the main connective of the formula $\neg(p \vee q)$ is $\neg$ (the negation), while the main connective of the fomrula $(\neg p \vee q)$, is $\vee$ (the disjunction). We call formulae whose main connective is $\wedge$ *conjunctions*. Similarly, if the main connective of a formula is a $\vee$, it is a *disjunction* and if the main connective is $\neg$, then it is a *negation*.

3.6 Showing That a Word Is Not in $\mathbb{PL}$

It is somewhat more difficult (or rather more verbose) to show that a word is not in $\mathbb{PL}$.

If the word is not over the right alphabet, such as the three-symbol word $\mathtt{p} + \mathtt{q}$, which uses a foreign symbol $+$, then we can easily dismiss it as $\mathbb{PL}$ only contains words over L.

However, if the word is over the right alphabet and we want to show that it is not part of $\mathbb{PL}$, we must make use of the minimality condition. Here is an example showing that $(\mathtt{p}\neg\mathtt{q}) \notin \mathbb{PL}$:

Let's assume (by contradiction) that $(\mathtt{p}\neg\mathtt{q}) \in \mathbb{PL}$. Then, by the minimality condition, it follows that $(\mathtt{p}\neg\mathtt{q}) \in \mathbb{PL}$ must be explained by one of the rules Base Case, Inductive Case i – iii.

But we cannot apply the Base Case to show that $(\mathtt{p}\neg\mathtt{q}) \notin \mathbb{PL}$, since $(\mathtt{p}\neg\mathtt{q}) \notin A$.

But we cannot apply the Inductive Case i either, because there is no formula φ such that $(\mathtt{p}\neg\mathtt{q}) = \neg\varphi$ (no matter how we would choose $\varphi \in \mathbb{PL}$, the first symbol of the word on the lhs would be $($, while the first symbol of the word on the rhs would be $\neg$).

But we cannot apply the Inductive Case ii either, because there are no formulae $\varphi_1, \varphi_2 \in \mathbb{PL}$ such that $(\mathtt{p}\neg\mathtt{q}) = (\varphi_1 \vee \varphi_2)$ (no matter how we would choose $\varphi_1, \varphi_2 \in \mathbb{PL}$, the word on the lhs would not contain the symbol $\neg$, while while the the word on the rhs would contain it).

By similar reasons, we cannot apply Inductive Case iii either.

But this contradicts our assumption and therefore it must be the case that $(\mathtt{p}\neg\mathtt{q}) \notin \mathbb{PL}$.

3.7 Unique Readability

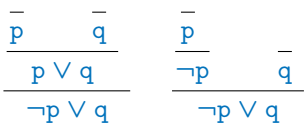
The definition of $\mathbb{PL}$ has an important property called *unique readability*:

Theorem 13 (Unique Readability of Propositional Formulae)**.** *For any word $\varphi \in \mathbb{PL}$, there is a unique construction tree.*

The property above is also sometimes called unambiguity of the grammar. Proving the Unique Readability Theorem is beyond the scope of the present lecture notes. Instead we will try to understand the importance of the property above by showing how an alternative, fictive, definition of the set $\mathbb{PL}$ would be ambiguous:

Beginning of Wrong Definition of $\mathbb{PL}$.

> Imagine that the Inductive Case ii would read:
>
> $\vdots$
>
> 3. (Inductive Step ii) If $\varphi_1, \varphi_2 \in \mathbb{PL}$, then $\varphi_1 \vee \varphi_2 \in \mathbb{PL}$.

> ⋮
>
> The only difference would be that we do not place parantheses around disjunctions. With this alternative, fictive, definition of $\mathbb{PL}$, we have that the word $\neg p \vee q \in \mathbb{PL}$. However, this word has two different construction trees:
>
> 
>
>
> Such an ambiguity would be very troubling, because we would not know if $\neg p \vee q$ is a disjunction (if we would use the construction tree on the right) or a negation (the construction tree on the left). In fact, avoiding such syntactic ambiguities was the main reason why we left natural language and began studying formal logic.

End of Wrong Definition of $\mathbb{PL}$.

I hope that you can now see that the Unique Readability Theorem is non-trivial (as an exercise to the more mathematically inclined students, I suggest that you try to prove it) and that it is a very important property of our definition of formulae, since it essential says that any propositional formula can be read unambiguously.

3.8 Object-language and meta-language

A peculiarity in logic is that we study propositions (we analyze, for example, their truth values), but in order to perform such study, we use a form of reasoning that also uses propositions. For example, in our study, we could make the following argument:

> *The proposition I do not go to school is false if the proposition I go to school is true.*

Note that the sentences that are the object of our study (*I do not go to school* and *I go to school*) are propositions, but so is the entire statement *The proposition I do not go to school is false if the proposition I go to school is true*. In this way, it seems that we are performing a circular reasoning, in which we study our own method of study.

This apparent difficulty does not occur in the case of arithmetic, for example. In arithmetic, we study numbers and operations over numbers, and we reason about numbers using propositions.

The difficulty can be solved by strictly separating *the object language* from the *meta-language*. The object language is the language that we study ($\mathbb{PL}$), and the meta-language is the language that we use to perform the study (English).

> The object language is the language that represents the object of our study (in our case, propositional logic). The meta-language is the language that we use to communicate about the object of our study (in our case, English).

In this course, to more easily differentiate between the two, we make the following convention: all elements in the object language are written in sans-serif blue font (for example, $(\mathsf{p} \wedge \mathsf{q})$), and the statement in the meta-language are written using *a regular black font* (for example, *any atomic formulae is a formula*).

> It is extremly important to differentiate correctly between object language and meta-language. To this effect, the lecture notes use a typographic convention. The elements of the meta-language are written in regular black font. The elements of the object-language are writte as follows: $(\mathsf{p} \wedge \mathsf{q})$. For this reason, if you print out the lecture notes, please print them in color.

3.9 Exercise Sheet

Exercise 14. *Show that the following words are* propositional formulae *(i.e., element of $\mathbb{PL}$), by explaining which are the construction steps (base case, respectively one of the three inductive steps):*

1. $\neg\mathsf{q}$;
2. $(\mathsf{p}_1 \wedge \mathsf{q})$;
3. $\neg(\mathsf{p} \vee \mathsf{q})$;
4. $(\neg\mathsf{p} \vee \neg\mathsf{q})$;
5. $\neg(\neg\mathsf{p} \vee (\mathsf{q} \wedge \neg\mathsf{q}))$.

Exercise 15. *Show that the following words over the alphabet L are not elements of $\mathbb{PL}$(hint: show that none of the four construction rules applies):*

1. $((\neg)\mathtt{q})$;
2. $\mathtt{q} \wedge \neg$;
3. $\mathtt{pq}$;
4. $\mathtt{p} \wedge \mathtt{q}$;
5. $(\mathtt{p}\neg\mathtt{q})$;
6. $(\mathtt{p}) \wedge (\mathtt{q})$.

Exercise 16. *Which of the following are formulae (in $\mathbb{PL}$) and which are not:*

1. $\mathtt{p}_1$;
2. $\mathtt{p}_1 \vee \mathtt{q}_1$;
3. $(\mathtt{p}_1 \vee \mathtt{q}_1)$;
4. $(\neg\mathtt{p}_1 \vee \mathtt{q}_1)$;
5. $((\neg\mathtt{p}_1) \vee \mathtt{q}_1)$;
6. $(\neg\mathtt{p})$?

Exercise 17. *Give 5 examples of interesting formulae (with many variables/operators, etc.).*