

# Format for describing CFGs

We describe this format by means of an example. Consider the following grammar:

```
S -> aSb
S ->
```

This grammar has a single non-terminal symbol **S**, two terminal symbols **a** and **b**, and two productions: the first one rewrites **S** into the word **aSb**, and the second one into the empty word. It is easy to see that the grammar generates the language  $\{a^n b^n \mid n \geq 0\}$ . Since both productions have the same left-hand side, they could be written in a single line like this:

```
S -> aSb |
```

It is also possible to introduce more non-terminals, if this is considered necessary or convenient. For instance, the following grammar is equivalent to the previous example:

```
S -> aXb |
X -> aSb |
```

We now give a more precise definition of the format. Grammars are described by a list of productions. A production starts with a non-terminal symbol, followed by  $\rightarrow$ , followed by the list of right-hand sides of such non-terminal separated by bars  $|$ . Each of the right-hand sides is a (possibly empty) list of symbols. A non-terminal symbol is an uppercase latin character, and a terminal symbol is any other character. We assume that the start symbol of the grammar is the first non-terminal to appear.