5.1 Sprachen und ihre Regeln

Tippen Sie in Ihren Taschenrechner `3 + × 4`. Viele Modelle antworten mit *Syntax ERROR*: Die Zeichenfolge ist gar keine Rechnung, der Rechner kann sie nicht einmal lesen. Tippen Sie dagegen `5 ÷ 0`, kommt *Math ERROR*: Die Eingabe ist tadellos gebaut, nur ergibt sie keinen Sinn. Ihr Taschenrechner unterscheidet damit genau die zwei Ebenen, um die es in diesem Abschnitt geht.

Sprachen dienen der Verständigung. Man unterscheidet **natürliche Sprachen**, die zwischen Menschen gewachsen sind (Deutsch, Französisch), und **künstliche Sprachen**, die jemand bewusst festgelegt hat. Programmiersprachen gehören dazu, aber auch die Formelsprache der Mathematik.

Jede Sprache gehorcht Regeln auf zwei Ebenen:

- Die **Syntax** legt die erlaubte Form von Wörtern und Sätzen fest.
- Die **Semantik** bestimmt deren Bedeutung.

Der Unterschied lässt sich am Beispiel zeigen. Der Satz „Der Hund ist ein Insekt" ist syntaktisch einwandfrei und semantisch falsch. Die Zeile `x := "drei" + 5;` ist umgekehrt semantisch klar gemeint und syntaktisch bzw. typmäßig unzulässig. Ein Compiler prüft die Syntax vollständig, die Semantik nur zu einem kleinen Teil — deshalb erzeugt ein fehlerfrei übersetztes Programm noch lange nicht das gewünschte Ergebnis.

Dass natürliche Sprachen erheblich komplizierter sind als Programmiersprachen, zeigt schon ein einziger Satz:

> Die deutsche Umgangssprache ist, wie jeder, der sie — und das wird an diesem Satz hier, den ich der Erläuterung wegen einschiebe, deutlich — lernen muss, weiß, eine sehr komplizierte Sprache.

Ein Mensch versteht ihn nach zweimaligem Lesen. Eine Maschine bräuchte dafür eine Grammatik, die beliebig tief verschachtelte Einschübe erlaubt — genau die Art von Struktur, an der die Automaten dieses Kapitels scheitern werden (Abschnitt 5.10).

Arbeitsauftrag 1

1. Nennen Sie je zwei natürliche und zwei künstliche Sprachen und begründen Sie die Zuordnung.
2. Geben Sie ein Beispiel für einen deutschen Satz, der syntaktisch korrekt, aber semantisch sinnlos ist — und eines für den umgekehrten Fall.
3. Ihr Programm übersetzt fehlerfrei, liefert aber falsche Ergebnisse. Welche Ebene ist betroffen? Begründen Sie.