

4. Spezifikation in PVS (II): Funktionen und Prädikate

Zur Erinnerung: Funktionstypen haben die Form $[arg\text{-typen} \rightarrow res\text{-typ}]$

Deklaration bzw. Definition von Funktionen:

- uninterpretiert:

$$f : [int \rightarrow int]$$

- interpretiert, d.h. mit einer *Definition*:

$$f(x: int): int = x * x$$

f wird hierdurch definiert als eine Funktion vom Typ $[int \rightarrow int]$ mit formalem Parameter x .

Eine alternative Notation für dieselbe Definition ist die *Lambda-Notation* (wird im folgenden Abschnitt eingeführt).

Der dem = folgende Ausdruck stellt den Rumpf der Funktion dar.

Rekursive Funktionen: kommen später

Prädikate und Mengen

Ein Prädikat ist einfach eine Funktion mit Wertebereich (= Wertetyp) `bool`:

`P : [int -> bool]`

z.B.

`P(x: int): bool = (x > 0)`

Alternative Notation in PVS:

`pred[T]` als Synonym für den Typ `[T -> bool]`

Ein Prädikat (als boolesche Funktion) kann als die *charakteristische Funktion* einer Menge angesehen werden, nämlich der Menge derjenigen Elemente des Definitionsbereichs, die das Prädikat erfüllen.

z.B. für die Menge der ganzen Zahlen zwischen 5 und 10:

`P(x: int): bool = (5 <= x AND x <= 10)`

Prädikate und Mengen (2)

Die Beziehung zwischen Prädikaten und Mengen (als Teilmengen einer Basismenge) ist eineindeutig.

In PVS verschwindet der konzeptionelle Unterschied zwischen Prädikaten und Teilmengen weitgehend.

Speziell: Mengen werden standardmäßig durch Prädikate (d.h. Funktionen) repräsentiert

~> s. unten