

Aufgabe 5-1

- Definieren Sie in PVS den Typ `even` der geraden natürlichen Zahlen als Subtyp von `nat`.
- Definieren Sie eine rekursive Funktion `half`, die eine gerade natürliche Zahl halbiert.
- Überprüfen Sie mit `M-x tc` ihre Theorie auf Typkorrektheit. Lassen Sie sich mit `M-x show-tccs` die entstandenen Typkorrektheitsbedingungen anzeigen. Es treten unterschiedliche Arten von TCCs auf: welche? Erklären Sie, warum diese TCCs generiert wurden. Beweisen Sie die TCCs mit `M-x pr`.
- Zeigen Sie:

```
half_halves : THEOREM
  FORALL (n:nat): half(2*n) = n
```

Aufgabe 5-2

Über einem Grundtyp `T` definieren wir Sequenzen mit fester Länge `N` als Funktion folgenden Typs:

```
Sequenz : TYPE = [below(N) -> T]
```

- Definieren Sie ein Prädikat `permutation_of?(S1,S2)`, das angibt, ob die Sequenz `S1` eine Permutation der Sequenz `S2` ist.
- Zeigen Sie, dass `permutation_of?` eine Äquivalenzrelation ist.
Hinweis: Eigenschaften bijektiver (injektiver / surjektiver) Funktionen aus der PVS-Prelude-Theorie können verwendet werden.
- Definieren Sie ein Prädikat `in?(x:T,S:Sequenz)` das angibt, ob ein Element `x` in einer Sequenz `S` enthalten ist.

Zeigen Sie:

```
perm_in? : LEMMA
  permutation_of?(S1,S2) => (in?(x,S1) = in?(x,S2))
```

Die Definition von `in?` kann auf zwei Arten geschehen – welche? Welche erscheint Ihnen für den Beweis günstiger?

