

Prolog: Prädikate auf Listen

Bearbeiten Sie die folgenden Aufgaben:

1. Schreiben sie ein Prädikat `myMember`, das der Typ-Spezifikation

`myMember(+Number, +List(Number))`

entspricht. Der Aufruf

`myMember(x, l)`

soll genau dann erfolgreich sein, wenn die Zahl x in der Liste l auftritt.

Bei allen weiteren Aufgaben sollen Sie zunächst Gleichungen aufstellen, die das Verhalten der zu implementierenden Funktionen beschreiben.

2. Schreiben Sie ein Prädikat `intersect` das der Typ-Spezifikation

`intersect(+List(Number), +List(Number), -List(Number))`

entspricht. Der Aufruf

`intersect(l1, l2, L)`

soll für zwei Listen l_1 und l_2 eine Liste l berechnen, die alle die Elemente enthält, die sowohl in l_1 als auch in l_2 auftreten.

3. Schreiben Sie ein Prädikat `small`, das mit der Typ-Spezifikation

`small(+Number), +List(Number), -List(Number)`

verträglich ist. Der Aufruf

`small(x, l, S)`

soll für eine Zahl x und eine Liste von Zahlen l die Liste aller der Zahlen aus l berechnen, die kleiner oder gleich x sind.

4. Schreiben Sie ein Prädikat `big`, das mit der Typ-Spezifikation

`big(+Number), +List(Number), -List(Number)`

verträglich ist. Der Aufruf

`big(x, l, S)`

soll für eine Zahl x und eine Liste von Zahlen l die Liste aller der Zahlen aus l berechnen, die größer als x sind.

5. Schreiben Sie ein Prädikat `quick_sort`, das mit der Typ-Spezifikation

`quick_sort(+List(Number), -List(Number))`

verträglich ist. Der Aufruf

`quick_sort(l, L)`

soll die Liste l sortieren.

Das Prädikat `quick_sort` soll nach der *divide-and-conquer*-Methode arbeiten:

- (a) Teilen Sie die zu sortierende Liste l zunächst mit den Prädikaten `small/3` und `big/3` in zwei Listen s und b auf. Die Liste s soll dabei alle Elemente enthalten, die kleiner als das erste Element der Liste l sind, während die Liste b die Elemente aus l enthält, die größer als das erste Element von l sind.
- (b) Sortieren Sie die Listen s und b .
- (c) Fassen Sie die beiden sortierten Listen zu einer sortierten Liste zusammen. Benutzen Sie dazu das Prädikat `concat` aus der Vorlesung.