

## Praktikum Spezifikation und Verifikation

### Aufsummieren einer Integer-Liste

Definieren Sie eine Funktion

**consts**

```
addlist :: "int list  $\Rightarrow$  int"
```

die als Argument eine Liste von ganzen Zahlen nimmt und alle Zahlen in der Liste aufaddiert.

Beweisen Sie:

1.  $addlist [1, 2, 3, 4] = 10$ ,
2.  $addlist [a, b, c, a, b, b] = 2 * a + 3 * b + c$ ,
3.  $addlist (a @ b) = addlist a + addlist b$ ,
4.  $addlist (rev a) = addlist a$ .

### Das Pascalsche Dreieck

Die ersten 8 Zeilen des *Pascalschen Dreiecks* lauten:

$n = 0$					1														
$n = 1$					1		1												
$n = 2$					1		2		1										
$n = 3$					1		3		3		1								
$n = 4$					1		4		6		4		1						
$n = 5$					1		5		10		10		5		1				
$n = 6$					1		6		15		20		15		6		1		
$n = 7$					1		7		21		35		35		21		7		1

Die inneren Elemente der  $n+1$ -ten Zeile im Pascalschen Dreieck werden konstruiert, indem man jeweils die Summe zweier benachbarter Elemente der  $n$ -ten Zeile bildet.

Benutzen Sie die Funktion

**consts**

```
zipadd :: "int  $\Rightarrow$  int list  $\Rightarrow$  int list"
```

**primrec**

```
"zipadd x [] = [1]"
```

```
"zipadd x (y#ys) = (x+y)#(zipadd y ys)"
```

um das Pascalsche Dreieck in Isabelle zu konstruieren. Die  $n$ -te Zeile des Dreiecks soll dabei durch eine Funktion mit folgender Signatur berechnet werden:

**consts**

```
row :: "nat  $\Rightarrow$  int list"
```

Beweisen Sie, dass  $row\ n$  für  $n = 0, 1, 2, 7$  genau die Werte in der obigen Abbildung berechnet. Beachten Sie dabei, dass Isabelle alle Zahlen ausser 0 und 1 binär kodiert. Lassen Sie sich von Isabelle  $row\ 10$  ausrechnen.

Beweisen Sie  $length\ (row\ n) = Suc\ n$ .

**TIP:** Beweisen Sie zuerst  $length\ (zipadd\ x\ xs) = length\ xs + 1$ .

**TIP:** Wenn Sie beim Beweis des obigen Hilfslemmas Schwierigkeiten bekommen, versuchen Sie stattdessen zu beweisen:  $\forall x. length\ (zipadd\ x\ xs) = length\ xs + 1$ .

Beweisen Sie:

1.  $row\ n \neq []$
2.  $last\ (row\ n) = 1$
3.  $addlist\ (row\ n) = 2 \wedge n$ .

▷ **Abgabe: 20. 4. 2005**