

## Übungen zur Rekursion

Implementiere in der Datei `lists.hs` eine Funktion `mylength(liste)`, die die Länge der übergebenen Liste berechnet. Die Funktion `myconcat(liste1, liste2)` soll als Ergebnis die Konkatenation der beiden Listen liefern. Mit `myappend(liste, elem)` soll das Element an das Ende der Liste angehängt werden. Die Funktion `listSum(liste)` soll die Summe aller Werte in der Liste zurückliefern. Verwenden Sie in dieser Aufgabe keine spezialisierten Listenfunktionen (wie z.B. den `++`-Operator) außer dem `:`-Operator.

```
1 myLength :: [a] -> Int
2 myLength [] = 0
3 myLength (x:xs) = 1+myLength(xs)
4
5 myconcat :: ([a],[a]) -> [a]
6 myconcat ([],ys) = ys
7 myconcat (xs,[]) = xs
8 myconcat ((x:xs),ys) = x : myconcat(xs,ys)
9
10 myappend :: ([Int],Int) -> [Int]
11 myappend ([],y) = [y]
12 myappend ((x:xs),y) = x:myappend(xs,y)
13
14 mySum :: [Int] -> Int
15 mySum [] = 0
16 mySum (x:xs) = x+mySum(xs)
```