

Übungsblatt 1

Aufgabe 1. Installieren Sie GHC (Glasgow Haskell Compiler, <https://www.haskell.org/ghc/>).

Aufgabe 2. Geben Sie jeweils einen Typ zu folgenden Haskell-Ausdrücken an, falls möglich.

(a) `10`

(b) `1 : []`

(c) `[1, "x"]`

(d) `f` mit `f x = x + 1`

(e) `f` mit `f g x = (g x) + 1`

Aufgabe 3. Auf <https://hoogle.haskell.org/> kann man nach Typen und Funktionen suchen, die in Haskell selbst oder auch in externen Bibliotheken definiert sind. Geben Sie die Definitionen folgender Typen bzw. Funktionen an.

(a) `String`

(b) `Bool`

(c) `filter`

(d) `div`

(e) `mod`

Aufgabe 4. Für `b :: Bool` und `x, y :: t` können Sie die Syntax `if b then x else y`

verwenden. Implementieren Sie folgende Funktionen:

(a) `isPowerOf2 :: Int -> Bool`,

wobei `isPowerOf2 x = True`, wenn $x \geq 1$ und es ein $y \in \mathbb{N}$ gibt mit $2^y = x$. In allen anderen Fällen soll `isPowerOf2 x = False` gelten.

(b) `filterPowersOf2 :: [Int] -> [Int]`,

wobei `filterPowersOf2 l` genau die Elemente `x` von `l` behält, für die `isPowerOf2 x = True` gilt. Die Reihenfolge der Elemente von `l` soll beibehalten werden.

(c) `filter :: (a -> Bool) -> [a] -> [a]` aus Aufgabe 3 c. Nennen Sie Ihre Implementierung `filter'`, da `filter` in Haskell bereits existiert.