

Informatik A – WS 1999/2000
2. Test Gruppe A 4.2.2000 (60 Minuten)

Tragen Sie alle Antworten und Lösungen möglichst auf diesem Blatt ein. Vermerken Sie bitte explizit, wenn Sie die Rückseite oder ein Zusatzblatt benutzen.

Name	Vorname	Matrikelnr.	Tutor

Aufgabe 1: (4 Punkte)

Gegeben ist der Blockcode $c : \{a, b, \dots, h\} \rightarrow \mathbb{B}^3$, der die ersten acht Buchstaben nach Tabelle codiert. Es sei $\bar{c} = c_p^2$ die Doppelcodierung mit Paritätsbit, d.h. $\bar{c}(x) = c(x)c(x)b_p(x)$ ($x \in \{a, \dots, h\}$). Decodieren Sie die folgenden Strings mit der 1-fehlerkorrigierenden Decodierung von \bar{c} und streichen Sie alle Codes durch, die nicht decodierbar sind.

$x \in \{a, \dots, h\}$	$c(x)$	1-fehlerkorrigierende Decodierung
<i>a</i>	000	0011010 →
<i>b</i>	001	
<i>c</i>	010	0111100 →
<i>d</i>	011	
<i>e</i>	100	1011011 →
<i>f</i>	101	
<i>g</i>	110	1101000 →
<i>h</i>	111	

Aufgabe 2: (4 Punkte)

Definieren Sie eine Funktion `merge :: [Int] -> [Int] -> [Int]`, die zwei aufsteigend sortierte Listen zu einer aufsteigend sortierten Liste vereinigt.

Bsp.: `merge [3,4,4,7,8] [1,2,4,8,9] ~> [1,2,3,4,4,4,7,8,8,9]`

Aufgabe 3: (3 Punkte)

Definieren Sie *in einer Zeile* eine Funktion `averageAge :: [(String,Int)] -> Int`, die für eine Liste von Personen mit Alter das Durchschnittsalter (ganzzahlig nach unten gerundet) berechnet. Eine korrekte, aber mehrzeilige Definition wird mit zwei Punkten bewertet.

Aufgabe 4: (3 Punkte)

Berechnen Sie den Huffman-Code für das folgende Alphabet mit Wahrscheinlichkeitsverteilung: $\{(a, 0.19), (b, 0.18), (c, 0.07), (d, 0.21), (e, 0.25), (f, 0.1)\}$

Schreiben Sie den berechneten Code in die folgende Tabelle und zeichnen Sie den Huffman-Baum darunter oder auf der Rückseite. (Nebenrechnung auf Rückseite oder Extrablatt, spielt bei der Bewertung keine Rolle).

Symbol	a	b	c	d	e	f
Codierung						

Aufgabe 5: (4 Punkte)

Geben Sie den Wert der folgenden Haskell-Ausdrücke an. Inkorrekte Ausdrücke sind zu streichen (ohne Begründung).

`foldr1 (++) ["ab ", ['a', 'd'], "", "cb"] ~`

`map length [['x', 'y'], "", "a c", 'x'] ~`

`foldr max 'u' "abtdg" ~`

`filter not [True, False, True] ~`

Aufgabe 6: (3 + 3 Punkte)

Bei einer Party stehen als Getränke nur Wodka und Orangensaft und Mischungen daraus zur Verfügung (auch Mischungen können gemischt werden):

```
data Drink = Wodka Int |           -- Parameter = Mengenangabe
            Orange Int |          -- Parameter = Mengenangabe
            Mix Drink Drink       -- Zusammengießen
```

a) Definieren Sie die Funktion `wodka :: Drink -> Int`, welche die Wodkamenge in dem Getränk bestimmt.

b) Sie können `wodka` und `orange :: Drink -> Int` (analog wie in a) verwenden. Definieren Sie den Typ `Drink` als Instanz der Klasse `Eq`, wobei zwei Getränke als gleich gelten, wenn sie das gleiche Mischungsverhältnis haben, wie z.B. `Mix (Wodka 4) (Orange 6) == Mix (Orange 9) (Wodka 6)`