# Übungsblatt 1

Florian

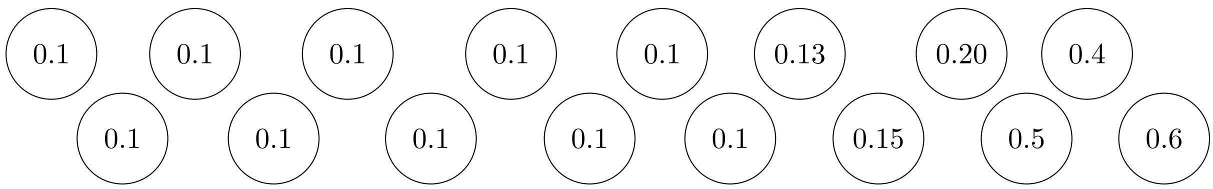
September 24, 2012

## Aufgabe 1

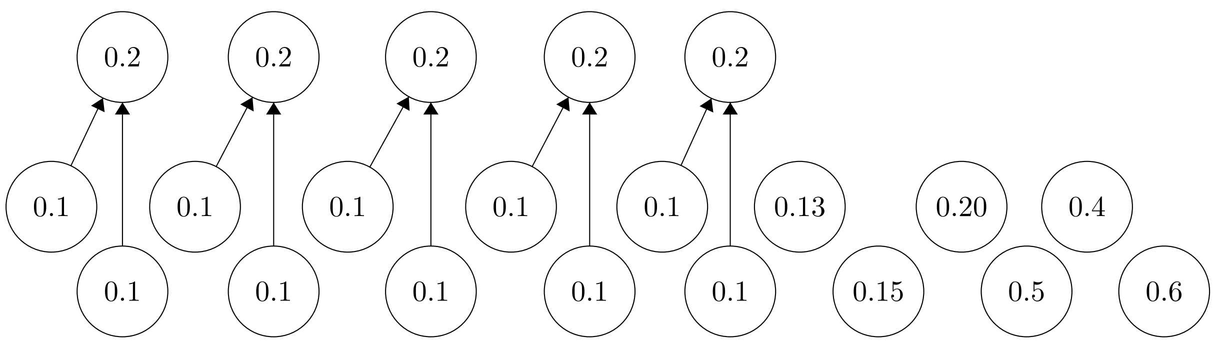
a) Die Auftretenswahrscheinlichkeiten $p(x)$ (wobei $\sum\limits_{x \in A} p(x) = 2.98$) werden normiert auf 1.0.

$$\begin{aligned}
E(A) &= \sum_{x \in A} p(x) \cdot \log_2 \frac{1}{p(x)} \\
&= 10 \cdot \frac{0.1}{2.98} \cdot \log_2 \frac{2.98}{0.1} + \frac{0.13}{2.98} \cdot \log_2 \frac{2.98}{0.13} + \frac{0.15}{2.98} \cdot \log_2 \frac{2.98}{0.15} \\
&\quad + \frac{0.20}{2.98} \cdot \log_2 \frac{2.98}{0.20} + \frac{0.5}{2.98} \cdot \log_2 \frac{2.98}{0.5} + \frac{0.4}{2.98} \cdot \log_2 \frac{2.98}{0.4} + \frac{0.6}{2.98} \cdot \log_2 \frac{2.98}{0.6} \\
&= 3.60565808338531\ldots
\end{aligned}$$

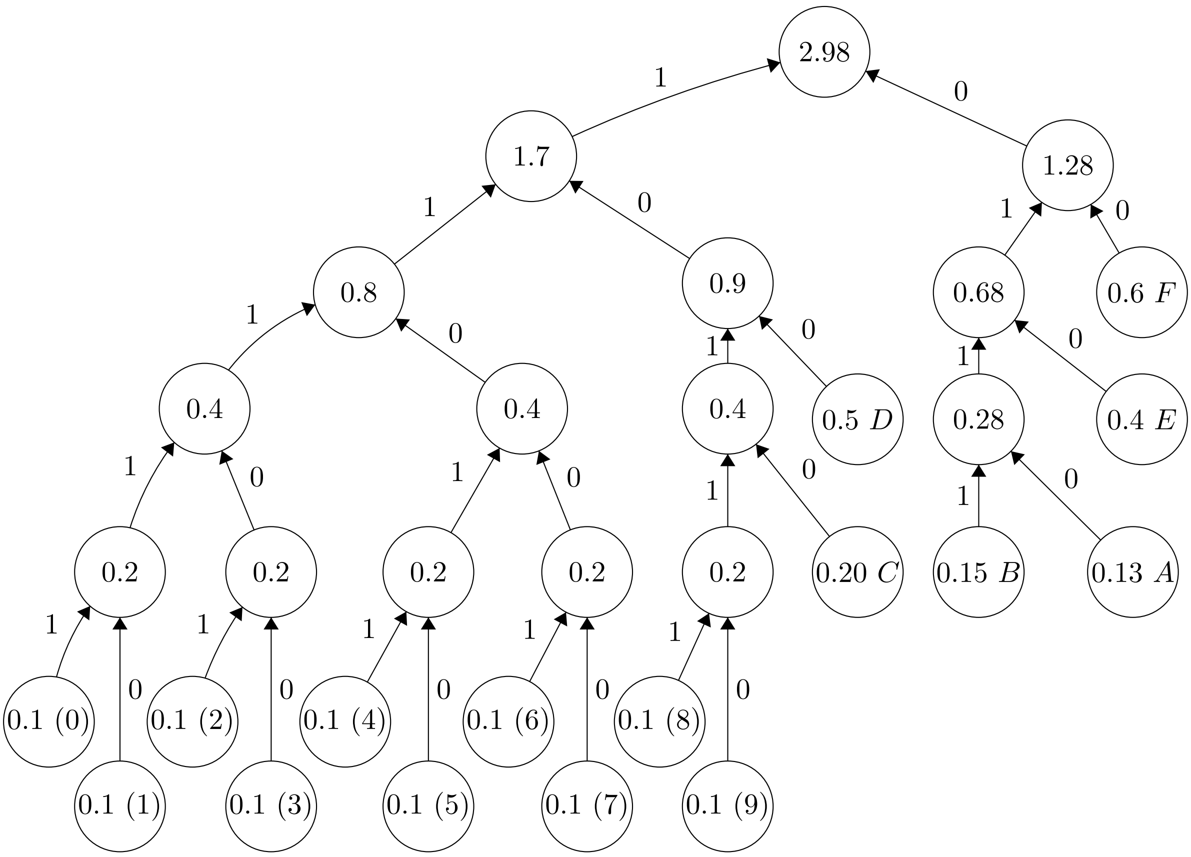
b) Nur Blätter:




Die 10 0.1-er zusammengefasst:



Und so weiter, und so fort bis zu:




Daraus folgt entlang der Pfeile der Code:

**0** 11111

**1** 11110

**2** 11101

**3** 11100

**4** 11011

**5** 11010

**6** 11001

**7** 11000

**8** 10111

**9** 10110

**A** 0110

**B** 0111

**C** 1010

**D** 100

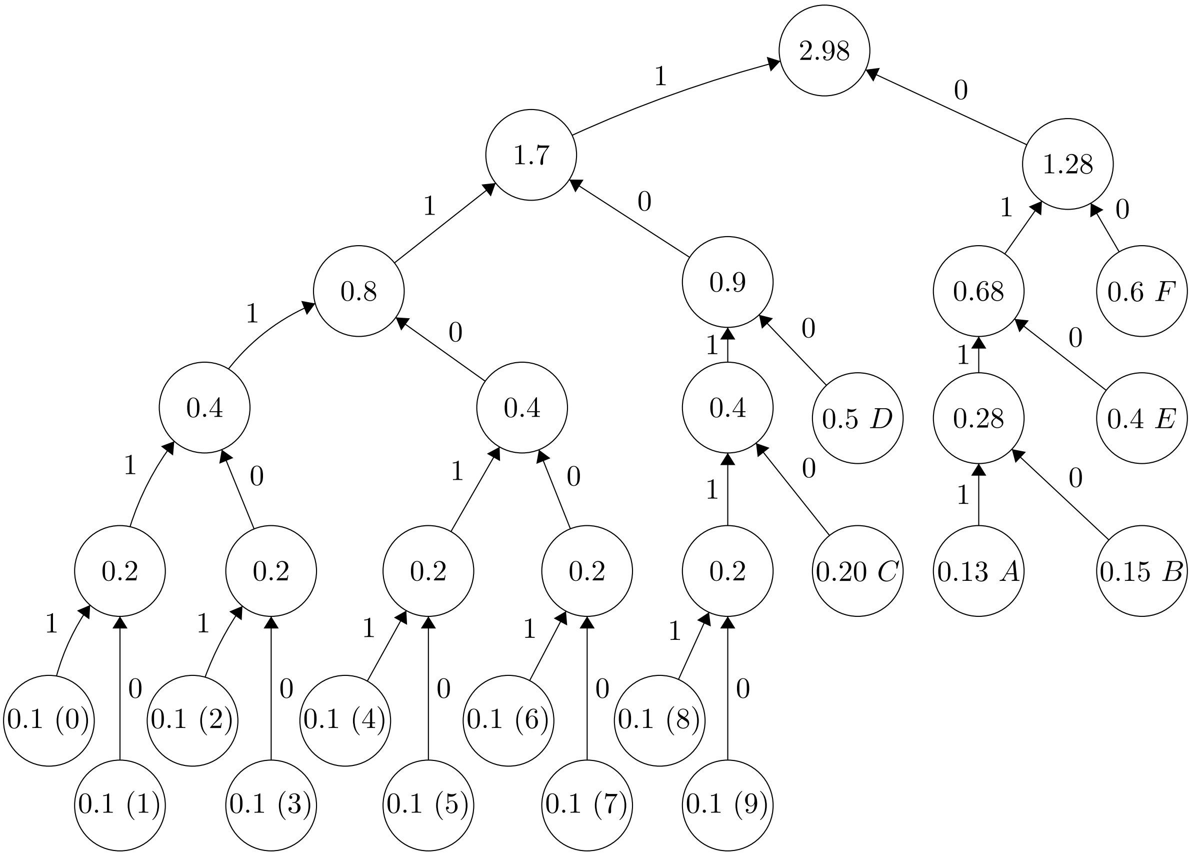
**E** 010

**F** 00

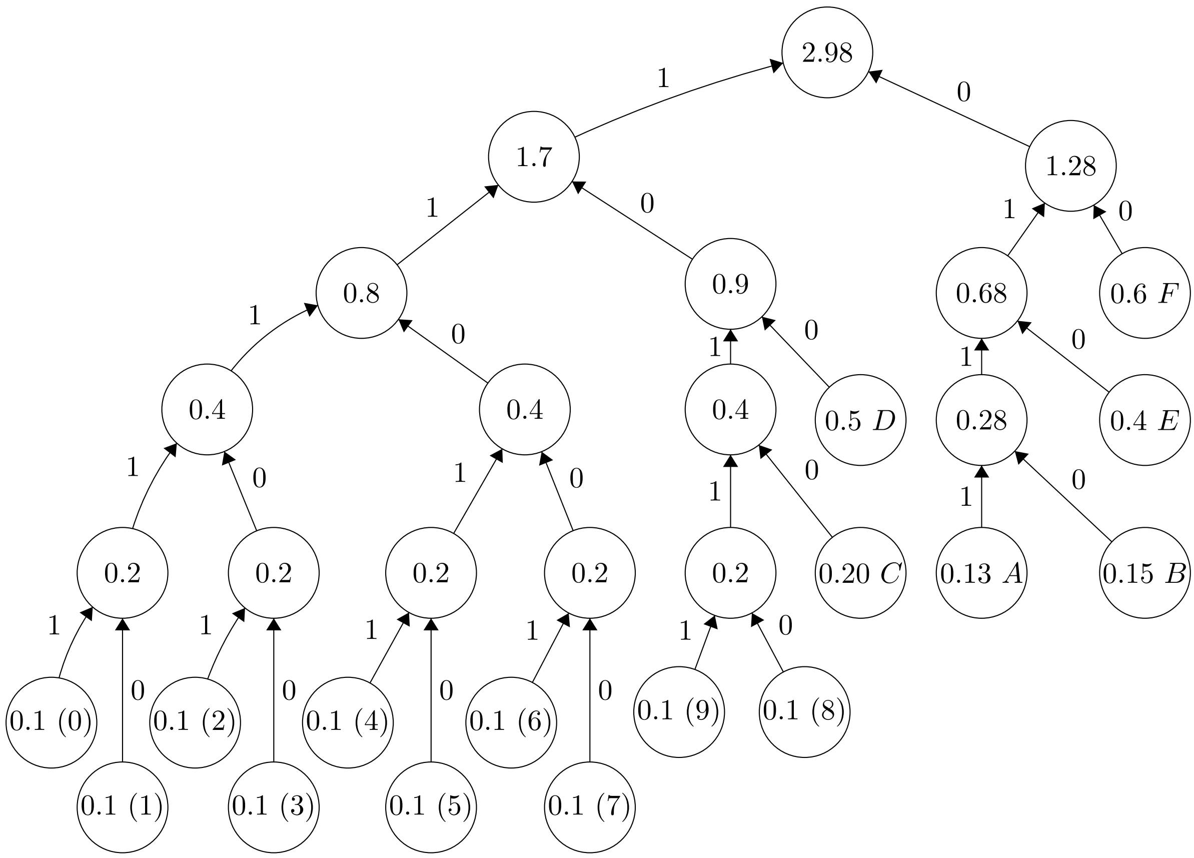
Die Code-Länge ist 5 (entspricht der Tiefe des Baums).

c) Der Baum ist nicht eindeutig.

Alternative Darstellungen:



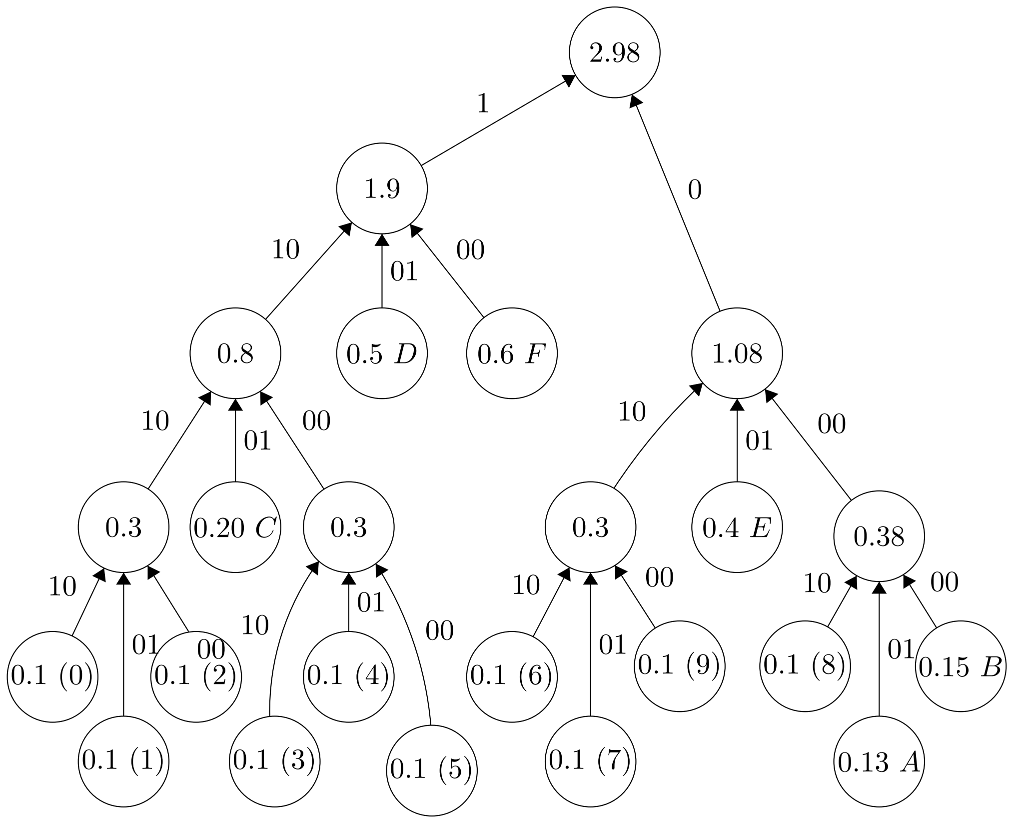

(A und B vertauscht)




(8 und 9 vertauscht)

d) Indem kein Binär-, sondern ein regulärer Baum mit beispielsweise 3 Kindkoten pro Knoten verwendet wird.

e) Regulärer 3-kindknotiger Baum:




f) Nein, weil auch die einzelnen Zeichen von Binärdaten wiederkehrende Häufigkeiten aufweisen (0 wird beispielsweise relativ häufig auftauchen).

## Aufgabe 2

1. Man kann Redundanz in die übertragenen Daten einbauen und dann die überschüssige Information verwenden, um Fehlererkennung bzw. -korrektur zu betreiben.

2. Die Kompression sollte vor der Chiffrierung erfolgen, weil das Ziel der Chiffrierung eine möglichst zufällige Verteilung der Zeichen ist. Dies hat negative Auswirkungen auf die Kompression, weil diese sich normalerweise wiederauftretende Muster bzw. ungleiche Auftretenswahrscheinlichkeiten zu Nutze macht. Die Fehlerkorrektur kommt sinnvollerweise erst am Ende, damit während dem Empfangen gleich als erstes die Fehleranalyse gemacht werden kann und nicht erst die Daten dechiffriert und dekomprimiert werden müssen.

   Also: Kompression ⇒ Chiffrierung ⇒ Fehlerkorrektur beim Senden, Fehlerkorrektur ⇒ Dechiffrierung ⇒ Dekompression beim Empfangen.

3. • Generatormatrix:

$$G = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

• Allgemein:

$$\begin{bmatrix} p_1 & p_2 & d_3 & p_4 & d_5 & d_6 & d_7 \end{bmatrix} = \begin{bmatrix} d_3 & d_5 & d_6 & d_7 \end{bmatrix} G$$

Mögliche Codewörter:

| $d_3d_5d_6d_7$ | $p_1 = d_3 + d_5 + d_7$ | $p_2 = d_3 + d_6 + d_7$ | $p_4 = d_5 + d_6 + d_7$ | $p_1p_2d_3p_4d_5d_6d_7$ |
|---:|---|---|---|---|
| 0000 | 0 | 0 | 0 | 0000000 |
| 0001 | 1 | 1 | 1 | 1101001 |
| 0010 | 0 | 1 | 1 | 0101010 |
| 0011 | 1 | 2=0 | 2=0 | 1000011 |
| 0100 | 1 | 0 | 1 | 1001100 |
| 0101 | 2=0 | 1 | 2=0 | 0100101 |
| 0110 | 1 | 1 | 2=0 | 1100110 |
| 0111 | 2=0 | 2=0 | 3=1 | 0001111 |
| 1000 | 1 | 1 | 0 | 1110000 |
| 1001 | 2=0 | 2=0 | 1 | 0011001 |
| 1010 | 1 | 2=0 | 1 | 1011010 |
| 1011 | 2=0 | 3=1 | 2=0 | 0110011 |
| 1100 | 2=0 | 1 | 1 | 0111100 |
| 1101 | 3=1 | 2=0 | 2=0 | 1010101 |
| 1110 | 2=0 | 2=0 | 2=0 | 0010110 |
| 1111 | 3=1 | 3=1 | 3=1 | 1111111 |

• Indem wir die übertragenen Codewörter mit der Checkmatrix $H$ multiplizieren und 0 als Resultat erwarten.

$$H = \begin{bmatrix} 1 & 0 & x & 0 & x & x & x \\ 0 & 1 & x & 0 & x & x & x \\ 0 & 0 & x & 1 & x & x & x \end{bmatrix}$$