



Prof. Dr. Rolf Drechsler, drechsler@informatik.uni-bremen.de, MZH 3510
Dr. Robert Wille, rwille@informatik.uni-bremen.de, MZH 3485
Eleonora Schönborn, eleonora@informatik.uni-bremen.de, MZH 3350

6. Übungsblatt zur Vorlesung Technische Informatik 1

Aufgabe 1

(8 Punkte)

Im Folgenden sollen Zeichen effizient kodiert werden, um eine möglichst hohe Kompressionsrate zu erzielen. Verwendet wird dazu der Huffman-Code. Eine (repräsentative) Stichprobe der zu kodierenden Daten dient der Ableitung des Codes.

a) Als Stichprobe dient der folgende Satz:

UM REKURSION ZU VERSTEHEN MUSS MAN ZUNAECHST REKURSION VERSTEHEN

Generiere für diesen Text den Huffman-Code, wobei Leerzeichen ignoriert werden. Damit die Generierung des Codes eindeutig ist, sind noch einige Verabredungen notwendig:

1. Sortiere die Zeichen von links nach rechts absteigend nach ihren Häufigkeiten und bei gleicher Häufigkeit in lexikographischer Reihenfolge (z.B. K,O statt O,K).
2. Beim Zusammenfassen zweier „Knoten“ sollen immer die am weitesten links stehenden Knoten gewählt werden, die für eine Verschmelzung in Frage kommen.
3. Bei der Zuweisung der Codebits bekommt immer der linke Nachfolger eine ‘1’, der rechte eine ‘0’ zugewiesen.
4. Ein neuer Knoten wird immer oberhalb seines linken Kindes angelegt.

Gib in deiner Lösung sowohl den Code für jeden einzelnen Buchstaben als auch den kodierten Text an. Trenne dabei zur Übersichtlichkeit die einzelnen kodierten Worte durch Leerzeichen.

b) Dekodiere mit Hilfe des in Teilaufgabe a) generierten Codes folgende Nachricht:

010001101101111011000011001011011111

c) Gib die Kompressionsrate ((Länge der unkodierten Nachricht)/(Länge der kodierten Nachricht)) für die Nachricht aus b) an, wenn ein Zeichen im unkodierten Fall durch 8 Bit repräsentiert wird.

Aufgabe 2

(8 Punkte)

- a) Überführe die Wörter 00110000_2 , 11011000_2 und 10000111_2 (dabei steht das Bit mit der höchsten Nummer links) in den in der Vorlesung vorgestellten 1-Bit-fehlerkorrigierenden Hamming-Code.
- b) Bei den kodierten Wörtern 001001010010_2 , 111011000110_2 und 100111100000_2 wurde jeweils ein Bit falsch übertragen. Finde heraus, um welches Bit es sich handelt.
- c) Kann es sich bei dem Fehler im ersten Wort aus b) auch um einen Mehrfach-Übertragungsfehler handeln (Begründung und ggf. Beispiel nicht vergessen)?
- d) Unter welchen Bedingungen lassen sich Mehrfach-Übertragungsfehler mit dem gegebenen Kodierungsschema korrekt erkennen?

Aufgabe 3

(4 Punkte)

Beweise das folgende Lemma:

Sei $A = \{a_1, \dots, a_k\}$ ein Alphabet der Größe k und sei $c : A \rightarrow \{0, 1, 2, \dots, r-1\}^n$ ein Code der festen Länge n , dann gilt

$$n \geq \lceil \log_r k \rceil.$$

Abgabetermin: vor Beginn der Vorlesung am 7. Juni 2012