Datenkompression (Rote), SS 2003, 1. Übungsblatt

Abgabe: Aufgabe 1 und 2: bis zum Donnerstag, 24.4.2003, in der Vorlesung
Aufgabe 3: bis zum Montag, 28.4.2003, im Sekretariat (Raum 111)

1. (10 Punkte) Wie viele verschiedene Nachrichten der Länge n lassen sich aus den Codewörtern aba, $aaaab$, bb, ba bilden? Jedes Codewort darf dabei beliebig oft verwendet werden. Geben Sie eine Rekursionsformel für die Anzahl A_n der Nachrichten der Länge n an.

2. (10 Punkte) Man kann Dateien mit einer Länge von vier Bit auf folgende Weise komprimieren. Wenn die Datei nur aus Nullen besteht, wird sie unverändert abgespeichert. Andernfalls wird die erste Eins, zusammen mit allen eventuell davor stehenden Nullen, entfernt und nur der Rest abgespeichert.

   Auf diese Weise werden fast alle 4-Bit-Dateien (mit einer einzigen Ausnahme) um mindestens 1 Bit verkürzt.

   (a) (5 Punkte) Zeigen Sie, dass man aus den komprimierten Dateien die ursprünglichen Dateien eindeutig wiederherstellen kann.

   (b) (5 Punkte) Berechnen Sie die durchschnittliche Länge der entstehenden komprimierten Dateien unter der Annahme, dass alle 4-Bit Dateien gleich wahrscheinlich sind.

   (c) Diskutieren Sie, ob diese Methode geeignet verallgemeinert werden kann, sodass es für ein praktisches Kompressionsverfahren in Frage kommt.

3. (10 Punkte) Welche der folgenden Codes sind präfixfrei? Welche sind eindeutig entzifferbar?

   (a) (2 Punkte) 001, 0100, 110, 10, 111.

   (b) (2 Punkte) 001, 0100, 110, 010, 111, 01.

   (c) (6 Punkte) b, ba, ba^7, ba^{13}, ba^{14}; a^3b, a^3ba^2, a^3ba^4, a^3ba^6; a^8b, a^8ba^2, a^8ba^4, a^8ba^6; $a^{11}b$, $a^{11}ba$, $a^{11}ba^2$. (a^{14} bedeutet 14 Wiederholungen des Buchstabens a, usw.)

4. (0 Punkte) Gegeben sei eine Alphabet $S = \{a_1, \ldots, a_n\}$ mit n Symbolen und eine Wahrscheinlichkeitsverteilung $p = (p_1, \ldots, p_n)$ mit Entropie $H(p)$. Rechen Sie nach, dass das Alphabet S^2, in dem das „Symbolpaar“ a_ia_j mit Wahrscheinlichkeit p_ip_j vorkommt, Entropie $2H(p)$ hat.

5. (0 Punkte) Finden Sie den Fehler im folgenden „Beweis“, dass die Kraft'sche Ungleichung hinreichend für die Existenz eines präfixfreien Codes ist.

   > Gegeben seien die Längen $l_1, l_2, \ldots, l_n$, die die Kraft'sche Ungleichung erfüllen. Die maximale Länge sei $h = l_{\max}$. Wir beginnen mit einem vollständigen binären Baum der Höhe h mit 2^h Blättern. Jeder Knoten des Baumes auf Tiefe t entspricht einem 0-1-Wort der Länge t. Für das erste Codewort wählen wir einen Knoten der Tiefe l_1. Dadurch wird der Teilbaum, der unter diesem Knoten hängt, für zukünftige Codewörter gesperrt; es werden somit 2^{h-l_1} Blätter eliminiert. Als nächstes wird ein verbleibender Knoten auf Tiefe l_2 ausgewählt und dadurch 2^{h-l_2} Blätter eliminiert, und so weiter. Man kann dabei niemals stecken bleiben, denn aus der Kraft'schen Ungleichung ergibt sich, dass bei der Wahl des k-ten Wortes noch mindestens
   >
   > $$2^h - (2^{h-l_1} + 2^{h-l_2} + \cdots + 2^{h-l_{k-1}}) \geq 2^{h-l_k}$$
   >
   > Blätter übrig sind.