Universität Ulm - Abteilung Angewandte Informationsverarbeitung
Prof. Dr. Franz Schweiggert und Hans Braxmeier, Sommersemester 2003

9. Übungsblatt zur Vorlesung Allgemeine Informatik II

Abgabetermin: Donnerstag, 17.07.2003

Aufgabe 1: Datenkompression *(10 Punkte)*

Den etwas wirren Astronomieprofessor vom letzten Übungsblatt haben Sie sicher noch nicht vergessen...

Da, wie sei bereits wissen, der Sternenkatalog astronomisch groß ist und somit auf der Festplatte unheimlich viel Platz verbraucht, fragt der Professor Sie noch einmal um Rat. Diesmal sollen die Daten komprimiert werden!

In der Literatur finden sich dazu verschiedene Verfahren wie Huffman-Codeierung, Run-Length-Algorithmus und vieles mehr. Unter anderem ist auch ein Verfahren beschrieben, wie Daten mit dem Lempel-Ziv-Algorithmus (1978) komprimiert werden können!

Der Algorithmus sei im Folgenden vorgestellt:

Motivation und Eigenschaften

- Substituierender Kompressionsalgorithmus
- Verlustfreie Kompression
- Zur Dekompression werden keine zusätlichen Informationen benötigt (wie z.B. beim Huffman-Code)

Datenstrom und Aufbau des Baumes (Beispiel)

- Es liege folgender Datenstrom vor: ababcbababaaaaaaaEOF
- Hieraus wird der Baum, der am Ende des Übungsblattes gezeigt wird, aufgebaut.

Aufbau des Baumes im Detail

- Erzeuge den Wurzelknoten (im Bild Knoten 0)
- Lese erstes Zeichen (a) und hänge neuen Knoten ein (im Bild Knoten 1)

- Lese nächstes Zeichen (b). Schon im Baum vorhanden? Nein! Erzeuge neuen Knoten (im Bild Knoten 2)
- Lese nächstes Zeichen (a). Schon im Baum vorhanden? Ja! Folge diesem Ast bis zum Ende, lese nächstes Zeichen (b) und hänge an dieses Ende (also auf Blattebene) einen neuen Knoten an (im Bild Knoten 3)
- Speichere jeweils die Nr. des Astes sowie das nachfolgende Zeichen ab (siehe Ausgabestrom).
- Verfahre so weiter, bis alle Zeichen eingelesen sind.
- Der komprimierte Ausgabestrom hat am Ende folgendes Aussehen: 0a 0b 1b 0c 2a 5b 1a 7a 7EOF

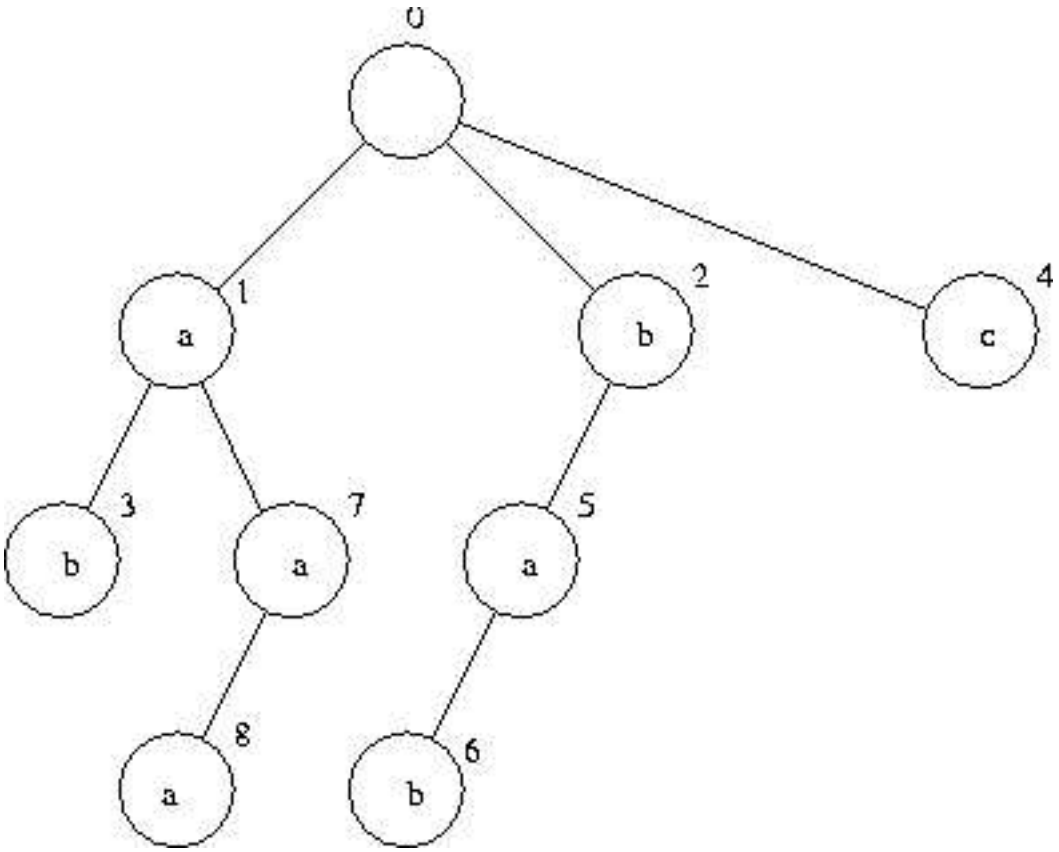
Die Dekomprimierung

- NUR MIT HILFE DES AUSGABESTROMS kann nun der Baum aufgebaut und der ursprüngliche Datenstrom wieder hergestellt werden!

Was ist zu tun?

- Gegeben sei ein beliebiger Datenstrom bestehen aus den Zeichen a, b und c.
- Erzeugen Sie nun mit Hilfe des oben genannten Algorithmus einen zu den Daten passenden Baum.
- Traversieren Sie den Baum in Post-Order und geben Sie den Inhalt aller Knoten aus!

Der Codebaum:



Viel Erfolg!