
Algorithmen für die Speicherhierarchie

Abgabetermin: 28.01.2008 vor der Übung

Strings

Aufgabe 1 (10 Punkte)

Die Malaufgabe:

Gegeben ist das Wort *Mississippi*. Zeichnen Sie einen Compact-Trie und einen Pat-Trie, die jeweils alle Suffixe des Wortes enthalten. Geben sie das Suffix- und das LCP-Array an.

(Falls die genaue Definition nicht mehr ganz klar sein sollte, hilft die angegebene Literatur oder auch Google.)

Aufgabe 2 (10 Punkte)

Gegeben ist ein Text der Länge N und das zugehörige Suffix/LCP-Array und ein Suchpattern P .

Geben Sie einen Intern-Speicher-Algorithmus an, der die Z verschiedenen Auftreten des Pattern im Text bestimmt. Welche Komplexität hat Ihr Algorithmus?

Finden Sie einen $O(\log N + |P| + Z)$ -Algorithmus?

Aufgabe 3 (10 Punkte)

In der Vorlesung wurde der String-B-Tree vorgestellt. Dabei handelt es sich um einen Baum, dessen Blätter Strings repräsentieren. Die internen Knoten nutzen Pat-Tries zum navigieren.

Warum nutzt man Pat-Tries anstatt von Compact-Tries?

Ist es nötig, die im Pat-Trie ausgelassenen Informationen in jedem Baumknoten zu überprüfen?

Aufgabe 4 (10 Punkte)

Am Ende der Vorlesung wurde der Doubling-Algorithmus zur Bestimmung des Suffix/LCP-Arrays aus einem Text vorgestellt. Welche Laufzeit hat der Algorithmus im besten/schlechtesten Fall?