Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Dr. Volker Heun

Wintersemester 2001/2002
Übungsblatt 5
21. November 2001

Algorithmische Bioinformatik

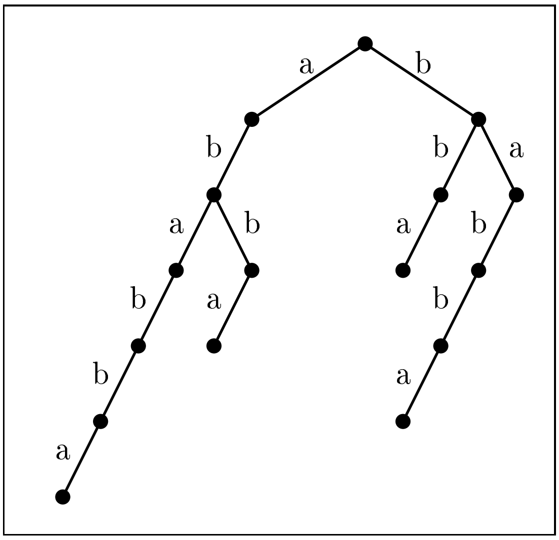
Abgabetermin: Donnerstag, den 29. November (15^{00} Uhr) in der Übung

Aufgabe 1

Auf der rechten Seite ist der Suffix-Trie für das Wort $t = ababba$ angegeben. Konstruiere aus diesem mit Hilfe des in der Vorlesung angegebenen On-Line-Algorithmus schrittweise die Suffix-Tries für

- $t' := t \cdot b = ababbab$,
- $t'' := t' \cdot a = ababbaba$ und
- $t''' := t'' \cdot a = ababbabaa$.

Man gebe hierzu für jedes Wort t', t'' und t''' eine eigene Zeichnung mitsamt der verwendeten und neu konstruierten Suffix-Links an.



Hinweis: *Die als freiwillig gekennzeichneten Aufgaben werden nicht bewertet und sind für die Klausurzulassung ohne Bedeutung.*

Freiwillige Aufgabe 1

a) Entwerfe einen Algorithmus für das Problem *exact string matching* auf der Basis von Suffixbäumen, der zu einem Text t und einem Suchwort s die Position *jedes* Vorkommens von s in t ausgibt. Für k Vorkommen von s in t soll die Komplexität des Algorithmus $O(|s| + |t| + k)$ betragen.

b) Wie kann man die Komplexität der Suche nach s auf $O(|s|)$ senken, wenn nur höchstens ein Vorkommen von s in t auszugeben ist?

c) Welchen Platz belegt eine optimal implementierte Datenstruktur für einen Suffixbaum eines Textes der Länge n?

Freiwillige Aufgabe 2

Gegeben sei eine Menge $S = \{s_1, \ldots, s_\ell\}$ von Zeichenreihen mit $n = \sum_{i=1}^{\ell} |s_i|$. Konstruiere einen Algorithmus, mit dem man in Zeit $O(n)$ alle Zeichenreihen $s_i \in S$ finden kann, die Teilwörter einer anderen Zeichenreihe $s_j \in S$ sind.