Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Ernst W. Mayr/Dr. Sven Kosub
Moritz Maaß

Sommersemester 2003/04
Übungsblatt 2
19. Mai 2004

Internet-Algorithmik

Abgabetermin: 19.05.2004 **vor** *der Vorlesung*

Aufgabe 1

In der Vorlesung wurde für die approximative Suche in Texten die Fehlerpropagationsregel für die Edit-Distanz mit den Regeln $\mathcal{E} = \{\mathsf{ins}, \mathsf{del}, \mathsf{sub}\}$ bestimmt.

(a) Bestimmen Sie die Fehlerpropagationsregel für die Typo-Distanz $\mathcal{T} = \{\mathsf{ins}, \mathsf{del}, \mathsf{sub}, \mathsf{tran}\}$, wobei die Operationen für *Einfügen* eine Buchstabens (ins), *Löschen* eines Buchstabens (del), *Ersetzen* eines Buchstabens (sub) und *Vertauschen* zweier benachbarter Buchstaben (tran) stehen.

(b) Wie sieht der Algorithmus zur Berechnung der Typo-Distanz $d_{\mathcal{T}}(v_1 \cdots v_m, u_1 \cdots u_m)$ aus?

Aufgabe 2

Zeigen Sie, dass die Version H der approximativen Suche in Hypertext für die Operationsmenge $\mathcal{E} = \{\mathsf{ins}, \mathsf{del}, \mathsf{sub}\}$ $\mathcal{NP}$-vollständig ist.

Aufgabe 3

In der Übung wurden folgende Definitionen vorgestellt.

Definition 1 (Azyklischer elementarer Hypertext) *Ein Tripel $H = (V, E, T)$ ist ein* elementar Hypertext*, falls $E \subseteq V \times V$ und $T : V \to \Sigma$ ist.*
Ist (V, E) ein azyklischer Graph, so heißt der Hypertext $H = (V, E, T)$ ebenfalls azyklisch.

Definition 2 (Geordneter Backtracking-Pfad) *Wir definieren geordnete Backtracking induktiv:*

- *Jeder gerichtete Pfad $(v_1, v_2, \dots, v_m)$ ist ein geordneter Backtracking-Pfad,*
- *Sei v ein Knoten mit $\{(v, u_1), (v, u_2), \dots, (v, u_k)\} \subset E$ mit einer Ordnung $(v, u_1) < (v, u_2) < \cdots < (v, u_k)$ auf den Kanten. Seien weiterhin $(u_1, u_{1,2}, \dots, u_{1,m_1})$, $(u_2, u_{2,2}, \dots, u_{2,m_2})$ bis $(u_k, u_{k,2}, \dots, u_{k,m_k})$ geordnete Backtracking-Pfade, dann ist auch $(v, u_1, u_{1,2}, \dots, u_{1,m_1}, u_2, u_{2,2}, \dots, u_{2,m_2}, \dots, u_k, u_{k,2}, \dots, u_{k,m_k})$ ein geordneter Backtracking-Pfad.*

Definition 3 (Hypertext Matching mit geordnetem Backtracking) *Ein Suchwort $P = p_1 \cdots p_m \in \Sigma^m$ kommt genau dann* mit geordnetem Backtracking *im Hypertext $H = (V, E, T)$ vor, wenn es ein geordneten Backtracking-Pfad $v_1, v_2, \dots, v_m$ mit $P = T(v_1)T(v_2) \cdots T(v_m)$ gibt.*

Geben Sie einen effizienten Algorithmus für die exakte Suche eines Suchwortes $P = p_1 \cdots p_m$ mit geordnetem Backtracking in einem elementaren azyklischen Hypertext an!

Aufgabe 4

Endliche Automaten sind ein einfaches algorithmisches Modell für die Verarbeitung von Datenströmen. Wir interessieren uns für die Pass-Komplexität bei Datenströmen, d.h. wie oft die Eingabe gelesen werden muss, um eine bestimmte Aufgabe zu lösen.
Wir definieren die Klasse der regulären k-Pass-Sprachen wie folgt:

$$\begin{aligned} k\text{-}\textsc{Pass-Reg} = \ & \{L \subseteq \Sigma^* \mid \text{es gibt eine reguläre Sprache } B \subseteq (\Sigma \cup \{\#\})^*, \\ & \text{so dass für alle } x \in \Sigma^* \text{ gilt: } x \in L \Leftrightarrow (x\#)^k \in B\} \end{aligned}$$

(a) Bestimmen Sie die Klasse 1-Pass-Reg.

(b) Zeigen Sie, dass 1-Pass-Reg $\subseteq$ 2-Pass-Reg gilt.

(c) Ist 2-Pass-Reg mehr als 1-Pass-Reg? D.h. gibt es ein $L \in$ 2-Pass-Reg mit $L \notin$ 1-Pass-Reg?