
Effiziente Algorithmen und Datenstrukturen II

Abgabetermin: 24. Mai 2006 vor der Zentralübung

Aufgabe 1 (10 Punkte)

Berechnen Sie die Tabelle der d -Werte im Knuth-Morris-Pratt Algorithmus für das „Wort“ *ACHABACHABLACHS*.

Aufgabe 2 (10 Punkte)

Berechnen Sie die Tabelle der D -Werte im Boyer-Moore Algorithmus für das „Wort“ *ULULLULAUL*.

Aufgabe 3 (10 Punkte)

Sei $s \in \Sigma^m$ ein String über einem endlichen Alphabet. Um die D -Tabelle des Boyer-Moore Algorithmus in Zeit $O(m)$ zu berechnen, benötigt man mehrere Zwischenschritte. Zunächst brauchen wir einen Algorithmus, der für s die p_j -Werte für alle $j \in \{1, \dots, m\}$ berechnet, wobei

$p_j =_{\text{def}}$ Länge des längsten Präfixes von $(s_j \dots s_m)$, das auch Präfix von s ist.

- (a) Geben Sie einen Algorithmus zur Berechnung der p_j -Werte an, der höchstens Laufzeit $O(m^2)$ hat. (*Hinweis:* Es existiert sogar ein Algorithmus, der die p_j -Werte in Zeit $O(m)$ berechnet.)
- (b) Für alle $j \in \{1, \dots, m\}$ sei

$f_j =_{\text{def}}$ Länge des längsten Suffixes von $(s_1 \dots s_j)$, das auch Suffix von s ist.

Geben Sie einen Linearzeitalgorithmus für die Berechnung aller f_j -Werte für den Fall an, dass Sie einen Linearzeitalgorithmus für die Berechnung der p_j -Werte haben.

- (c) Die D_i -Werte des BM-Algorithmus berechnen sich nun wie folgt aus den f_j 's:

$$D_i = \begin{cases} \max_{j < m} \{j \mid f_j(s) = m - i + 1\} & , \text{ falls ein solches } j \text{ existiert} \\ \max_{j < m-i+1} \{j \mid f_j(s) = j\} & , \text{ sonst.} \end{cases}$$

Geben Sie einen Linearzeitalgorithmus für die Berechnung der D_i -Werte an, für den Fall, dass Sie einen Linearzeitalgorithmus für die Berechnung der p_j -Werte haben.