Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Dr. Hanjo Täubig
Johannes Nowak

Sommersemester 2008
Übungsblatt 9
11. Juli 2008

Algorithmische Bioinformatik I

Aufgabe 1

Sei $s = s_1 \cdots s_n$ eine Zeichenreihe der Länge n. Ein k-mer t von s ist eine Teilwort der Länge k von s, d.h. es existiert ein $i \in [1 : n-k+1]$ mit $s_i \cdots s_{i+k-1} = t$. Für eine Zeichenreihe s sei $\mathcal{M}_k(s) = \{s_i \cdots s_{i+k-1} : i \in [1 : n-k+1]\}$ die Multimenge der k-mere von s.
Zeigen Sie, dass es zwei Zeichenreihe s und s' gibt, so dass $\mathcal{M}_4(s) = \mathcal{M}_4(s')$ ist. Dies bedeutet, dass man aus einer Menge der k-mere die ursprüngliche Zeichenreihe nicht mehr eindeutig rekonstruieren kann.
Wie sieht es für beliebiges k aus?

Aufgabe 2

Ein Suffix-Array S für einen String $t \in \Sigma^n$ ist ein Feld der Größe n mit folgenden Eigenschaften:

i) $\forall i < j \in [1 : n] : t_{S[i]} \ldots t_n < t_{S[j]} \ldots t_n$

ii) $\{S[i] : i \in [1 : n\} = [1 : n]$

Dabei vergleichen wir zwei Wörter s und t bzgl. ihrer lexikographischen Ordnung, also z.B. $aba < abb$. Ein Suffix-Array speichert also die Suffixe eines Wortes in lexikographischer Reihenfolge. Entwerfen Sie einen effizienten Algorithmus, der für ein gegebenes t ein Suffix-Array konstruiert.

Aufgabe 3

Für $x, y \in \Sigma^*$ wird eine Sequenz $z \in \Sigma^n$ als gemeinsame Oberfolge von x und y bezeichnet, wenn we zwei streng monoton wachsende Folgen $(i_1, \ldots, i_{|x|}) \in [1 : n]^{|x|}$ und $(j_1, \ldots, j_{|y|}) \in [1 : n]^{|y|}$ gibt, so dass $z_{i_1} \ldots z_{i_{|x|}} = x$ und $z_{i_1} \ldots z_{j_{|y|}} = y$.
Konstruieren Sie einen möglichst effizienten Algorithmzus, der für $x, y \in \Sigma^*$ eine kürzeste gemeinsame Oberfolge bestimmt. Analysieren Sie die Laufzeit.