

Algorithmen auf Sequenzen

Übungsblatt 9

Abgabetermin: Dienstag, 10.01.2017, 10 Uhr
(Via email an sophie.friedl@bio.ifi.lmu.de oder persönlich)

1. Aufgabe (Berechnung von LCP-Werten, *nicht abzugeben*)

Zeigen Sie, wie man für den in der Vorlesung vorgestellten Algorithmus zum Suchen in Suffix-Arrays für $t \in \Sigma^n$ mittels binärer Suche in Zeit $O(m + \log(n))$ die benötigten lcp-Werte in einem Preprocessing mit einem Zeitbedarf $O(n)$ und Platzbedarf $O(n)$ unabhängig vom Suchwort $s \in \Sigma^m$ vorab berechnen kann, so dass jede lcp-Anfrage bei der eigentlichen Suche nach s in konstanter Zeit beantwortet werden kann.

HINWEIS: Das Feld L mit $L[i] = \text{lcp}(i - 1, i)$ darf verwendet werden.

2. Aufgabe (Suche in Suffix-Arrays)

In der Vorlesung wurde gezeigt, wie man mit Hilfe eines Suffix-Arrays für $t \in \Sigma^n$ in Zeit $O(m + \log(n))$ feststellen kann, ob t ein Wort $s \in \Sigma^m$ enthält oder nicht. Modifizieren Sie diesen Algorithmus so, dass er alle z Vorkommen von s in t mit Zeitbedarf auf $O(m + \log(n) + z)$ findet.

Wie kann man den Zeitbedarf auf $O(m + \log(n))$ senken?

3. Aufgabe (Finden von Teilstrings)

Sei $t \in \Sigma^*$ ein Text und $k \in \mathbb{N}$. Wie kann in $O(|t|)$ Zeit festgestellt werden, wie viele verschiedene Teilstrings der Länge k in t enthalten sind?