

Übung zu Algorithmen auf Sequenzen Blatt 4

Ausgabe: 27. November 2014 **Besprechung:** 11. Dezember 2014

Aufgabe 4.1

Beschreibe, wie man aus dem Text t , dem Suffixarray `pos` und dem `lcp`-Array den Suffixbaum konstruieren kann. Welche elementare Datenstruktur ist dafür notwendig?

Aufgabe 4.2

Die Fibonacci-Strings F_n sind definiert durch

$$\begin{aligned}F_0 &:= \mathbf{a}, \\F_1 &:= \mathbf{b}, \\F_n &:= F_{n-2}F_{n-1} \text{ für } n \geq 2.\end{aligned}$$

Berechne F_6 und konstruiere schrittweise den Suffixbaum dazu mit Ukkonen's Algorithmus.

Aufgabe 4.3

Konstruiere das Suffixarray `pos` und das `lcp`-Array für den Text `CAACACAAA$`. Der Algorithmus kann dabei beliebig gewählt werden.

Aufgabe 4.4

Beschreibe, wie man einen Suffixbaum nutzen kann, um ein Muster in einem Text fehlertolerant zu finden. Fehlertolerant soll hier konkret bedeuten, dass an höchstens einer beliebigen Position der Buchstabe durch einen andere Buchstaben ersetzt werden darf (Substitution).

Beispiel: Lautet das Muster `anna`, so sollen auch auch Vorkommen von `anja` oder `anne` gefunden werden, aber nicht von `abba` (zwei Substitutionen) oder `ana` (eine Deletion).

Aufgabe 4.5

Gegeben ist eine nichtleere Menge $W = \{w_1, \dots, w_n\}$ von Wörtern über einem Alphabet Σ , sowie das Suffixarray und `lcp`-Array zu $t = w_1\$_1 \dots w_n\$_n$.

Gesucht ist die Länge ℓ des kürzesten Strings, der in keinem Wort aus W als Teilstring vorkommt. (Dieser String selbst muss nicht ausgegeben werden.)

Gibt einen Algorithmus an, der ℓ effizient berechnet.