Algorithmische Grundlagen und Vermittlung der Informatik
Lehrstuhl 11 (Algorithm Engineering)
Prof. Dr. Johannes Fischer

technische universität dortmund

Text Indexing and Information Retrieval

Übungsblatt 4

Besprechung: 14.11.2016

Aufgabe 1 (Praxis)

Suffix Arrays für natürlichsprachliche Texte können auch wortbasiert sein: sortiere nur die Textpositionen, an denen ein Wort beginnt (also z.B. nach jedem Whitespace und Satzzeichen). Implementieren Sie ein solches Verfahren (z.B. mit Hilfe Ihrer bisherigen Implementierungen oder der Implementierung von sais aus dem letzten Aufgabenblatt) und vergleichen Sie Platz- und Zeitbedarf mit "herkömmlichen" Suffix Arrays. Testen Sie *verschiedene* Textarten von `http://pizzachili.dcc.uchile.cl/texts.html`.

Aufgabe 2 (Praxis)

Man kann das LCP-Array H auf naive Art und Weise erstellen, indem man die Formel $H[i] = \max\{h \geq 0 : T[A[i], ..., A[i] + h - 1] = T[A[i - 1], ..., A[i - 1] + h - 1]\}$ für alle Werte $1 < i \leq n$ anwendet. Implementieren Sie das Verfahren und vergleichen Sie die Laufzeit mit dem Linearzeit-Algorithmus aus der Vorlesung auf *verschiedenen* Textarten von `http://pizzachili.dcc.uchile.cl/texts.html`. Versuchen Sie auch, einen 50MB großen Text zu generieren, auf dem der naive Algorithmus schneller ist.

Aufgabe 3 (Theorie)

Zeigen oder widerlegen Sie: wenn im LCP-Array der Wert ℓ (an einer beliebigen Stelle) auftritt, dann tritt auch der Wert $\ell - 1$ (an einer beliebigen anderen Stelle) auf.

Aufgabe 4 (Theorie)

Entwerfen Sie einen Linearzeitalgorithmus, der für einen Text T das *kürzeste* Teilwort findet, das nur einmal in T vorkommt. Hinweis: Suffixbäume!