

Algorithmen zur Sequenzanalyse: Übungsblatt 2

Enno Ohlebusch
Timo Beller

Das Übungsblatt wird am 12.11.2013 besprochen.

Für dieses Übungsblatt gelte $S_1 = \text{annaassannasananas\$}$. In Tabelle 1 ist das Suffix Array von S_1 gegeben.

Aufgabe 1

Berechnen Sie mit Hilfe des Kasai-Algorithmus das LCP-Array von S_1 . Implementieren Sie den Kasai-Algorithmus.

Aufgabe 2

Der Φ -Algorithmus berechnet aus einem String und dessen Suffix Array das LCP-Array. Im Gegensatz zum Kasai-Algorithmus wird jedoch das so genannte Φ -Array statt dem inversen Suffix Array benutzt. In Algorithmus 1 ist der Beginn des Φ -Algorithmus dargestellt. Vervollständigen Sie den Algorithmus, sodass das LCP-Array berechnet wird.

Aufgabe 3

Implementieren Sie den Φ -Algorithmus. Vergleichen Sie experimentell die Laufzeit des Φ -Algorithmus mit der Laufzeit des Kasai-Algorithmus. Auf der Vorlesungshomepage stehen hierfür Testdaten bereit. Geben Sie außerdem für jede Phase der beide Algorithmen an, auf welche Datenstrukturen Random-Access nötig ist.

Algorithmus 1 Der Φ -Algorithmus bekommt einen String S der Länge n und dessen Suffix Array SA übergeben. Achtung: Im Algorithmus wird von 0 an gezählt.

```
 $\Phi[0] \leftarrow SA[n - 1]$ 
for  $i \leftarrow 1$  to  $n - 1$  do
     $\Phi[SA[i]] \leftarrow SA[i - 1]$ 

 $\ell \leftarrow 0$ 
for  $i \leftarrow 0$  to  $n - 1$  do
     $j \leftarrow \Phi[i]$ 
    while  $S[i + \ell] = S[j + \ell]$  do
         $\ell \leftarrow \ell + 1$ 
     $\Phi[i] \leftarrow \ell$ 
     $\ell \leftarrow \max(0, \ell - 1)$ 
```

```
// ToDo
```

i	SA	$S_{\text{SA}[i]}$
1	19	\$
2	4	aassannasananas\$
3	13	ananas\$
4	15	anas\$
5	1	annaassannasananas\$
6	8	annasananas\$
7	17	as\$
8	11	asananas\$
9	5	assannasananas\$
10	3	naassannasananas\$
11	14	nanas\$
12	16	nas\$
13	10	nasananas\$
14	2	nnaassannasananas\$
15	9	nnasananas\$
16	18	s\$
17	12	sananas\$
18	7	sannasananas\$
19	6	ssannasananas\$

Tabelle 1: Das Suffix Array des Strings $S_1 = \text{annaassannasananas\$}$.