

3. Übungsblatt

Ausgabe: 27.04.2010 **Abgabe:** 04.05.2010, vor der Vorlesung

Aufgabe 1: Affixe

10 Punkte

- (a) Wie viele Präfixe hat das Wort INFORMATIK?
- (b) Wie viele Suffixe hat das Wort BIOLOGIE?
- (c) Wie viele Infixe hat das Wort INTERNET?
- (d) Wie viele nicht-leere Präfixe des Wortes $t =_{\text{def}} \text{AAABBBAAA}$ sind auch Suffixe von t ?
- (e) Geben Sie ein Wort an, so dass kein nicht-leerer Präfix, der echt kürzer als das ganze Wort ist, zugleich auch Suffix des Wortes ist.

Aufgabe 2: Exakte Suche in Texten

10 Punkte

Entwerfen Sie eine Variante des einfachen Algorithmus zur exakten Suche in Texten (mit Fenstervergleichen von links nach rechts), der alle Vorkommen eines Wortes s in einem Wort t findet. Implementieren Sie dazu in der Klasse `SearchClass` die Methode `searchAllStringsFromLeftToRight`:

```
public class SearchClass {  
    public static int[] searchAllStringsFromLeftToRight(String s, String t){  
        :  
    }  
}
```

Fügen Sie bitte auch erläuternde Kommentare ein.

Aufgabe 3: Algorithmus von Boyer und Moore

10 Punkte

Wie viele Vergleiche von Buchstaben benötigt der Algorithmus von Boyer und Moore (mit *bad character rule*), um das Wort TACAAG im Wort AACACTACACGTACAAGC zu finden? Verwenden Sie dazu die folgende Verschiebungstabelle:

	0	1	2	3	4	5	6
A	1	2	1	2	1	1	2
C	1	2	3	1	2	3	4
G	1	2	3	4	5	6	1
T	1	1	2	3	4	5	6

Dabei finden Sie zum Beispiel den Eintrag $S(3, G)$ in der Zeile G und der Spalte 3, d.h. $S(3, G) = 4$.

Veranschaulichen Sie auch den Ablauf des Algorithmus.