
Algorithmische Bioinformatik I

Abgabetermin: Mittwoch, den 19. Juni, vor der Vorlesung

Aufgabe (Notenbonus) 1

Ein Wort $w \in \Sigma^+$ besitzt die Periode $v \in \Sigma^+$, wenn $|v| \leq |w|$ gilt und wenn es ein $i \in \mathbb{N}$ gibt, so dass w ein Präfix von v^i ist; w besitzt dann die Periodenlänge $|v|$.

Konstruiere einen Algorithmus, der mit linear vielen Zeichenvergleichen die **kürzeste** Periode (bzw. Periodenlänge) eines gegebenen Wortes ermittelt.

Hinweis: Korrektheitsbeweis und Laufzeitanalyse nicht vergessen!

Aufgabe (Notenbonus) 2

Bestimme für das Wort $s = abaabbabaabbabab$ die Shift-Tabelle für den Boyer-Moore-Algorithmus aus der Vorlesung.

Gib dabei nicht nur die Tabelle, sondern auch alle Zwischenschritte für die Berechnung an (wie im Skript bzw. in der Vorlesung).

Aufgabe 3

Betrachte den unten angegebenen Boyer-Moore-Algorithmus, der für die Bestimmung der Shifts bei einem Mismatch nur die Extended-Bad-Character-Rule berücksichtigt. Gib eine Konstante $c > 0$ sowie eine unendliche Familie $\mathcal{F} = \{(s, t) : s, t \in \Sigma^*\}$ an, die für jedes Paar $(m, n) \in \mathbb{N}^2$ ein Paar $(s, t) \in \mathcal{F}$ mit $|s| \geq m$, $|t| \geq n$ und $|s| \leq |t|$ enthält und bei dem diese Variante für eine erfolglose Suche mindestens $c \cdot (|s| \cdot |t|)$ Zeichenvergleiche ausführt.

BC-Boyer-Moore (char $t[]$, int n , char $s[]$, int m)

```
int  $i := 0$ ,  $j := m - 1$ ;  
while ( $i \leq n - m$ ) do  
    while ( $t[i + j] = s[j]$ ) do  
        if ( $j = 0$ ) then return TRUE;  
         $j--$ ;  
     $i := i + j - \max\{k : (k < j \wedge s[k] = t[i + j]) \vee (k = -1)\}$ ;  
     $j := m - 1$ ;  
return FALSE;
```
