

4. Übung „Algorithmen der Bioinformatik I“

1. Implementieren Sie den *Boyer-Moore-Algorithmus* mit einfacher *Bad Character Rule* und dem *Strong Good Suffix Preprocessing*. Verwenden Sie dabei den in der Vorlesung vorgestellten Algorithmus zur Berechnung der $l'(i)$ in linearer Laufzeit. (5 Punkte)

2. Im Folgenden sollen die Laufzeiten des *naiven Algorithmus* (Übung 1), des *Z-Algorithmus* (Übung 2), des *naiven Algorithmus mit bad character rule* (Übung 3) und des *Boyer-Moore-Algorithmus* aus dieser Übung in Abhängigkeit von Alphabetgröße und Musterlänge untersucht werden.

Zur Analyse der Laufzeiten soll jeweils die Anzahl durchgeführter Vergleiche herangezogen werden. Inkrementieren Sie dazu einen Zähler vor oder nach **jedem** erfolgten Zeichenvergleich. Der Grund für diese (und keine möglicherweise effizientere) Zählung ist, dass wir nun die Operation *Zeichenvergleich und Zählerinkrementierung* als eine komplexere Vergleichsoperation betrachten können und somit die Laufzeiten für die verschiedenen Algorithmen in gleichem Maße beeinflusst werden. Außerdem soll die Laufzeit (Realttime oder Usertime, je nach Möglichkeit) gemessen werden.

Auf der Internetseite zur Vorlesung finden sich Texte über einem Alphabet von 40 bzw. 80 Zeichen. Suchen Sie in diesen beiden Texten jeweils nach Mustern der Längen 10, 20, 50, 100, 200, 500, 1000, 2000 und 5000. Erstellen Sie anschließend für beide Alphabetgrößen Diagramme, die die benötigte Anzahl an Vergleichen pro Musterlänge zeigen, und Diagramme, die die Laufzeit pro Musterlänge zeigen, jeweils in Abhängigkeit der Musterlänge. Was lässt sich daraus ersehen? Welcher Algorithmus braucht für jede Musterlänge am wenigsten Vergleiche? Welcher Algorithmus braucht real am wenigsten Zeit? (5 Punkte)

Abgabe des Programms per E-Mail an { gohr | grau } @ informatik.uni-halle.de.