Übungen zur Vorlesung
Algorithmen auf Sequenzen

TU Dortmund, SS 2008
Prof. Dr. Sven Rahmann

Blatt 8 vom 23.06.2008
Abgabe am Fr 27.06.2008 in der Vorlesung

Aufgabe 1 Wenn man von konvexen Gapkosten $g(k)$ spricht, meint man tatsächlich konkave Gapkosten: $g(k+1) - g(k) \leq g(k'+1) - g(k')$ für alle $k' \leq k$. Warum machen tatsächlich konvexe Gapkosten (mit $\geq$ statt $\leq$) keinen Sinn?

Aufgabe 2 Zum Selbststudium: die Four Russians - Methode.

1. Zeige: In der verwendeten DP-Matrix beim lokalen Sequenzalignment bei Scores $+1$ für Match, -1 für Mismatch und Indel unterscheiden sich horizontal / vertikal / diagonal benachbarte Zellen jeweils maximal um 1.

2. Die DP-Matrix lässt sich in $t \times t$-Blöcke aufteilen, und zwar so, dass je zwei Blöcke mit jeweils einer Zeile und Spalte überlappen. Die Zahlen in der letzten Zeile und Spalte sind eine Funktion der Zahlen in der ersten Zeile und Spalte und der jeweiligen Teilstrings der Eingabe-Sequenzen.

   Zeige, dass es auch genügt, die *Differenzen* zwischen den Werten in der ersten Zeile und Spalte zu kennen (und die Sequenzen), um die Differenzen zwischen den Werten in der letzten Zeile und Spalte zu berechnen. Wie viele verschiedene $t \times t$ Blöcke gibt es also für ein Alphabet der Größe σ?

3. Man kann auf die Idee kommen, alle Blöcke vorzuberechnen. Wie viel Speicherplatz benötigt dies? Was ergibt sich für die Laufzeit des DP-Algorithmus? Wie sollte man t wählen?

Aufgabe 3 Implementiere Ukkonen's Verbesserung von Sellers' Algorithmus für die approximative Teilstringsuche. Nimm an, dass die Sequenzen als Strings über $\{0, 1, 2, 3, \dots\}$ gegeben sind.

Hinweis zum Testen: Erzeuge eine lange zufällige DNA-Sequenz (Genom). Wähle eine zufällige Stelle aus. Kopiere (mit ein paar Fehlern) den entsprechenden Teilstring der Länge z.B. 100; nimm diesen als Muster. Suche das Muster im Genom mit verschiedenen Fehlerschranken.

Aufgabe 4 Implementiere nun die NFA-Methode für das gleiche Problem. Achte darauf, dass die Methode auch bei Strings länger als 32 bzw. 64 Zeichen richtig funktioniert, oder fange zu lange Muster ab.

Vergleiche die Laufzeit für verschiedene Fehlerschranken mit der von Ukkonen's Algorithmus.