

Prof. Dr. Knut Reinert
Rene Rahn
Kathrin Trappe
Kathleen Gallo
Thomas Krannich



Institut für Informatik
AG Algorithmische Bioinformatik

Algorithmen und Datenstrukturen in der Bioinformatik

Drittes Übungsblatt WS 12/13

Abgabe Freitag, 09.11., 15:00 Uhr

Name: _____ Übungsgruppe: A B C

Matrikelnummer: _____

Niveau I

Aufgabe 1: Needleman-Wunsch Algorithmus für DNA-Sequenzen

Berechnen Sie für die Sequenzen $S = \text{gtac}$ (horizontal) und $T = \text{agcta}$ (vertikal) ein globales Alignment nach dem Algorithmus von Needleman-Wunsch.

Ein Match trägt 3 zum Score des Alignments bei, ein Mismatch -5 und ein Space -2 .

- Berechnen Sie die *DP-Matrix*, und
- tragen Sie dabei jeweils *alle* Traceback-Kanten ein (also eventuell mehrere pro Zelle).
- Heben Sie *die beim Traceback benutzten* Kanten besonders hervor, und
- notieren Sie das resultierende *globale Alignment*.

Aufgabe 2: Needleman-Wunsch Algorithmus für Protein-Sequenzen

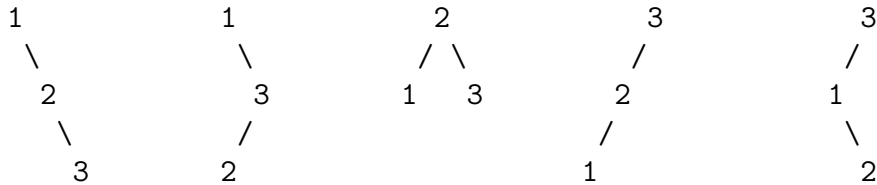
Berechnen Sie für die Sequenzen $S = \text{LILA}$ (horizontal) und $T = \text{RILKA}$ (vertikal) ein globales Alignment nach dem Algorithmus von Needleman-Wunsch.

- Suchen Sie im Internet nach der *PAM120-Matrix* (z.B. <http://www.behav.org/gene/bio-info/pam120.jpg>) und nehmen Sie zusätzlich eine Gap Penalty von -4 an.
- Führen Sie die Schritte a) bis d) wie bei Aufgabe 1 aus.

Niveau II

Aufgabe 3: Dynamische Programmierung

Wie viele verschiedene binäre Suchbäume mit den Schlüsseln $1 \dots n$ gibt es? Für $n = 3$ gibt es beispielsweise 5 verschiedene solcher Suchbäume:



- Geben Sie eine rekursive Berechnungsvorschrift der Zahl $T(n)$ an. Nutzen Sie $T(0) = T(1) = 1$ als Rekursionsanker.
- Skizzieren Sie einen Algorithmus, der dynamische Programmierung nutzt um $T(n)$ zu bestimmen.
- Was ist dessen Laufzeit?

Zur Erinnerung: Ein binärer Suchbaum ist ein echter Binärbaum (jeder Knoten hat genau zwei Kinder oder ist ein Blatt) mit der Eigenschaft, dass alle Elemente im linken Teilbaum eines Knotens x kleiner oder gleich x sind und alle Elemente im rechten Teilbaum größer oder gleich x sind.

Tipp zu (a): Jedes der n Elemente kann die Wurzel des Suchbaums sein und durch die Sucheigenschaft werden die restlichen Elemente auf die beiden Teilbäume darunter verteilt.