Algorithmen und Datenstrukturen 1

ALGO1 · SoSe-2021 · tcs.uni-frankfurt.de/algo1

⭐-Aufgabe: Sekundärstruktur

Revision 0e24027 (2021-06-28)

For an English version of this exercise, see [Erickson, page 149].

Ribonucleic acid (RNA) ist eine lange Kette von Millionen Nukleotiden oder *Basen* von vier verschiedenen Typen: Adenin (`A`), Cytosin (`C`), Guanin (`G`), und Uracil (`U`). Die *Sequenz* eines RNA-Moleküls ist eine Zeichenkette $b[1 \ldots n]$, wobei jedes Zeichen $b[i] \in \{\texttt{A}, \texttt{C}, \texttt{G}, \texttt{U}\}$ zu einer Basis korrespondiert. Zusätzlich zu den chemischen Verbindungen zwischen adjazenten Basen in der Sequenz können auch Wasserstoffbrückenbindungen zwischen manchen Basenpaaren entstehen. Die Menge der gebundenen Basenpaare heißt die *Sekundärstruktur* des RNA-Moleküls.

Wir sagen, dass zwei Basenpaare (i, j) und (i', j') mit $i < j$ und $i' < j'$ sich ***überlappen***, wenn $i < i' < j < j'$ oder $i' < i < j' < j$ gilt. In der Praxis überlappen sich die meisten Basenpaare nicht. Überlappende Basenpaaren bilden sogenannte Pseudoknoten in der Sekundärstruktur, die essenziell für bestimmte Funktionen der RNA sind, aber schwierig vorhergesagt werden können.

Wir wollen jetzt die bestmögliche Sekundärstruktur für eine gegebene RNA-Sequenz berechnen. Wir nehmen das folgende vereinfachte Modell der Sekundärstruktur an:

- Jede Basis kann mit höchstens einer weiteren Basis eine Bindung eingehen.
- Nur `A-U` Paare und `C-G` Paare können gebunden sein.
- Paare der Form $(i, i+1)$ und $(i, i+2)$ können sich nicht binden.
- Gebundene Basenpaare können sich nicht überlappen.

Diese letzte (und am wenigsten realistische) Einschränkung erlaubt es uns, die Sekundärstruktur der RNA als eine Art fetten Baum zu visualisieren:

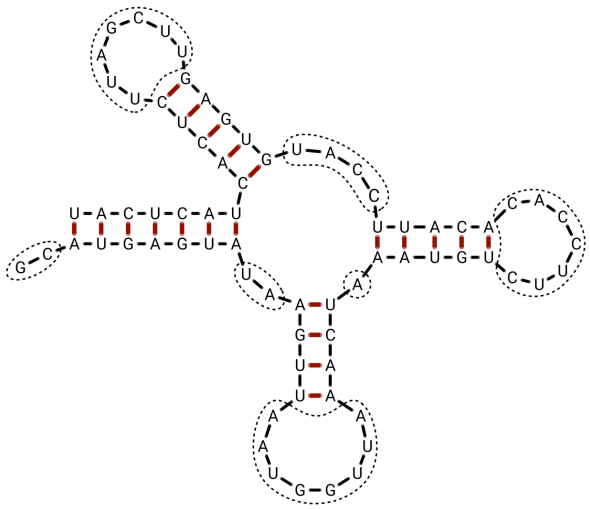

Abbildung 1: Beispiel einer RNA Sekundärstruktur mit 21 gebundenen Basenpaaren, die durch die fetten roten Linien gekennzeichnet sind. Lücken sind durch die gepunkteten Kurven gekennzeichnet. Diese Struktur hat eine Bewertung von $2^2 + 2^2 + 8^2 + 1^2 + 7^2 + 4^2 + 7^2 = 187$.

a) Beschreibe und analysiere einen Algorithmus, der die größtmögliche *Anzahl* an gebundenen Basenpaaren in der Sekundärstruktur einer gegebenen RNA-Sequenz berechnet.

b) Eine *Lücke* in der Sekundärstruktur ist ein maximaler Teilstring von ungebundenen Basen. Große Lücken sind chemisch instabil, daher sind Sekundärstrukturen mit kleineren Lücken viel wahrscheinlicher. Um diese Präferenz einzubeziehen, definieren wir jetzt die *Bewertung* einer Sekundärstruktur als die Summe der *Quadrate* der Längen aller Lücken; siehe die obige Abbildung. (Diese Bewertung ist fiktional; um die tatsächliche RNA-Struktur vorherzusagen braucht man *deutlich* komplizierte Bewertungsmethoden.)
Beschreibe und analysiere einen Algorithmus, der für eine gegebene RNA-Sequenz die kleinstmögliche Bewertung einer Sekundärstruktur berechnet.

Hinweise zur Abgabe. Im Buch [Erickson, Abschnitt 3.4] ist in kleinen Schritten beschrieben, wie man bei dynamischer Programmierung vorgeht. Folge diesen Schritten! Fang gar nicht erst an, über for-Schleifen nachzudenken, bevor du eine vollständige rekursive Lösung hast! Das beinhaltet eine klar verständliche deutsche oder englische Spezifikation der rekursiven Teilprobleme, die du löst (ohne diesen Teil erhältst du keinen ⭐). In der Algorithmenentwicklung, in der Programmierung, und in den meisten Aufgaben des Lebens gilt immer: **Mach es zuerst korrekt, dann effizient.** Den ⭐ erhälst du für die vollständige und weitgehend korrekte Bearbeitung der beiden Aufgabenteile.