Institut für Informatik
Praktische Informatik und Bioinformatik
Prof. Dr. Caroline Friedel
Marie-Sophie Friedl
Michael Kluge

Wintersemester 2016/17

Algorithmen auf Sequenzen

Übungsblatt 1

Abgabetermin: Montag, 31.10.2016, 10 Uhr
(Via email an sophie.friedl@bio.ifi.lmu.de oder persönlich)

Hinweise: Soweit nicht explizit anders angegeben, gibt es für alle Aufgaben 10 Punkte.

Die Übungsblätter können und sollen (soweit wie möglich) in Zweier-Teams bearbeitet werden. Bei elektronischer Abgabe müssen alle Dateien eindeutig nach dem Blatt bzw. der Aufgabe benannt werden, sowie der Nachname **beider** Teammitglieder im Dateinamen enthalten sein.

1. Aufgabe (MSS):

Maximal Scoring Subsequence (MSS)

Eingabe: Eine Folge $(a_1, \ldots, a_n) \in \mathbb{R}^n$ reeller Zahlen.
Ausgabe: Eine Teilfolge $(a_i, \ldots, a_j)$, die den maximalen Score $\sigma(i, j) = \sum_{\ell=i}^{j} a_\ell$ erzielt.

Geben Sie einen Algorithmus in Pseudo-Code an, der auf dem Algorithmus aus der Vorlesung für MSS mit dynamischer Programmierung basiert und abgesehen von dem Platz für die Eingabefolge mit konstantem Platz (zusätzlich zur Eingabe) und weiterhin mit quadratischen Zeitbedarf auskommt.

2. Aufgabe (MSS):

Sei $S > 0$ der Score einer *maximal scoring subsequence* einer Folge $a = (a_1, \ldots, a_n) \in \mathbb{R}^n$. Geben Sie einen möglichst effizienten Algorithmus an, der *alle* Teilfolgen mit Score S findet. Der Wert S ist natürlich a priori nicht bekannt.

Analysieren Sie den Algorithmus hinsichtlich Zeit- und Platzkomplexität.

Hinweis: Wenn für $i \leq k \leq \ell \leq j \in [1 : n]$ mit $(i < k)$ oder $(\ell < j)$ gilt, dass $\sigma(i, j) = \sigma(k, \ell) = S$, dann stellt $(a_i, \ldots, a_j)$ auch hier keine Lösung dar und ist nicht auszugeben.

Algorithmen für AMSS (All Maximal Scoring Subsequences) dürfen hier nicht verwendet werden.

3. Aufgabe (MSS):

Betrachten Sie das folgende Problem:

MAXIMAL SCORING SUBSEQUENCE WITH LOWER BOUND

Eingabe: Eine Folge $(a_1, \ldots, a_n) \in \mathbb{R}^n$ reeller Zahlen und eine natürliche Zahl $B \in \mathbb{N}$.

Ausgabe: Eine Teilfolge $(a_i, \ldots, a_j)$, die unter allen Teilfolgen der Länge mindestens B (d.h. $(j - i + 1) \geq B$) ihren Score $\sigma(i, j) = \sum_{\ell=i}^{j} a_\ell$ maximiert.

Konstruieren Sie für die Lösung dieses Problems einen Algorithmus mit linearem Zeitbedarf.