
Algorithmische Bioinformatik I

Abgabetermin: Donnerstag den 12. Mai, vor der Vorlesung

Aufgabe 1 (Programmieraufgabe)

Implementiere die Algorithmen für das *Maximal Scoring Subsequence Problem* basierend auf der dynamischen Programmierung, dem Divide-and-Conquer-Ansatz und der cleveren Lösung. Hierbei soll jeweils die kürzeste Lösung ausgegeben werden, die als erste in der Eingabefolge auftritt.

Beachten Sie folgende Vorgehensweise bei der **Implementierung**:

- Die Ausgabe für alle drei Varianten ist gleich.
- Start und Ende einer Lösung bilden ein Intervall.

Da die Ausgabe von allen drei Varianten gleich ist, empfiehlt es sich **wiederverwendbare** Funktionen zu überlegen. Beachten Sie dies beim Entwurf Ihrer Implementierung. Die Abgabe erfolgt über den Abgabeserver <https://services.bio.ifi.lmu.de/abgabeserver>.

Aufruf: `java -jar gruppenname_blatt02.jar [dc|dp|clever] "s0...sk"`

Das **erste** Argument $\in \{ \text{dc}, \text{dp}, \text{clever} \}$ gibt an, ob der Divide-and-Conquer-Ansatz, dynamische Programmierung oder der clevere Ansatz gewählt werden soll. Sollten Sie eine Variante noch nicht implementiert haben, lassen Sie die Ausgabe einfach leer.

Das **zweite** Argument ist eine Zahlenfolge, wobei zwei aufeinanderfolgende Zahlen durch ein **Leerzeichen** getrennt sind. Als Eingabe sind nur ganze Zahlen vorgesehen.

Die Ausgabe soll, zeilenweise getrennt, zunächst die identifizierte **MSS**, dann **Start** und **Ende** sowie die **Score** des MSS beinhalten. Beachten Sie, dass Sie die **erste** (am weitesten links vorkommende) kürzeste MSS ausgeben sollen.

Bitte vergessen Sie nicht, dass Ihre *jar*-Datei auch Ihren Source-Code enthalten muss!

Beispiel:

Sequenz: { 1 2 -5 1 3 5 -10 4 5 -2 -7 4 5 -1 }

Ausgabe:

4 5

7

8

9