



Allgemeine Hinweise

Ausgabe der Übung: 20.10.2015
Abgabe: 02.11.2015 bis 14 Uhr über Moodle oder Postfach
Testat am: 03.11.2015

- Die Abnahme der Übungen gilt als **Prüfungsleistung**. Bei einer Verhinderung durch Krankheit ist eine ärztliche Bescheinigung der **Arbeitsunfähigkeit** vorzulegen.
- Geben Sie Ihre Lösung **schriftlich (ausgedruckt) oder als PDF-Datei über Moodle** beim Dozenten ab. Benutzen Sie das unter <http://mi.informatik.hs-mannheim.de/gumbel/en/verschiedenes/templates/> vorgegebene Deckblatt für Übungen. Drucken Sie auf jede Seite die Seitenzahl und heften Sie Ihre Ausarbeitung.
- Zur Abnahme müssen die Ergebnisse am Computer demonstrierbar sein.
- Bei der Abnahme der Übung ist der **Studentenausweis** vorzulegen.
- Die Abnahme findet im **Pool-Raum** statt.

Hinweise zu den Aufgaben

Sofern nicht anders angegeben, sollen die Programme in Java, Scala oder Matlab geschrieben sein. Etwaige vorhandene Funktionen, die das Problem direkt lösen, dürfen **nicht** genutzt werden.

Lernziele

- Molekularbiologische Prozesse verstehen
- Dynamische Programmierung verstehen
- Sequenziermethoden verstehen

Aufgabe 1: Translation NR1H4

Der menschliche Kernrezeptor NR1H4 – ein Gen, das die Genexpression regelt – wird in eine rund 90.000 bp unreife mRNA transkribiert (siehe erster Eintrag in der FASTA-Datei im Moodle-Kurs unter *Unterlagen für Übungen/Kernrezeptor NR1H4.fasta*). Diese beinhaltet 11 Exons, die in dieser Datei gespeichert sind. **Hinweis:** Es handelt sich hier eigentlich um eine DNA-Sequenz, d.h. es wird T statt U benutzt.

Aufgabe 1.1: Reife mRNA-Sequenz

Erzeugen Sie hieraus eine reife mRNA-Sequenz (DNA-Sequenz genügt) und wandeln Sie diese in ein Protein (Primärstruktur) um. Das Protein soll in der Datei NR1H4-Protein.fasta gespeichert werden und den einbuchstabigen Aminosäure-Code verwenden.

Aufgabe 1.2: Untranslated Regions

Geben Sie an, wo sich die 5'- bzw. 3'-Untranslated-Region befindet.

Aufgabe 1.3: Splice-Alternativen

Welches Protein würde entstehen, wenn Intron 1 nicht entfernt würde? Das Protein soll in der Datei NR1H4-Protein-Variante.fasta gespeichert werden und den einbuchstabigen Aminosäure-Code verwenden.

Hinweise

Nutzen Sie Matlab und das FASTA-Dateiformat.



Aufgabe 2: Aufbewahrungskosten

Für die Sequenzierung eines Organismus⁴ wird für die Projektdauer von 12 Wochen (allgemein n Wochen) ein Primer benötigt, der leider sehr aufwändig und damit kostspielig aufzubewahren ist. Deshalb beschließt der Laborleiter, dass jede Woche der Primer beim Hersteller neu bestellt werden kann und dadurch das Lager „klein“ gehalten werden kann. Der Hersteller liefert den Primer nur in ganzzahligen Einheiten und das Sequenzierungslabor kann maximal 10 (allgemein u_{max}) Einheiten gleichzeitig aufbewahren. Zu Beginn des Projekts ist das Lager leer und am Ende (nach 12 Wochen) soll es ebenfalls wieder leer sein. Für die Sequenzierung werden jede Woche 2 Primer-Einheiten (allgemein u_c Einheiten) gleichmäßig verbraucht.

Angenommen, die Kosten betragen für

- das Auslösen einer Bestellung 10€ pro Quadratwurzel der Bestellmenge (allgemein $C_o\sqrt{u_o}$) und
- für die wöchentliche Aufbewahrung 1000€ pro Primer-Einheit zum Quadrat (allgemein $C_s \cdot u_s^2$).

Hierbei ist $C_o = 10$ ein Kostenfaktor einer Bestellung, u_o bezeichnet die Bestellmenge, $C_s = 1000$ ist ein Kostenfaktor für die aufzubewahrenden Primer-Einheiten und u_s bezeichnet den Lagerbestand. Der eigentliche Preis für eine Primer-Einheit soll hier keine Rolle spielen.

➡ Welche Bestellmengen sollte das Labor über die 12 Wochen ordern, um die Kosten zu minimieren?

Aufgabe 2.1: Dynamische Programmierung

Formulieren Sie dieses Problem für einen Dynamische-Programmierung-Algorithmus. Was sind die Zustände und welche Größe hat die Matrix M ? Geben Sie die Funktionen für die Entscheidungen d , Werte v und Vorgängerzustände p an. In welchem Zustand sind die minimalen Gesamtkosten zu finden?

Aufgabe 2.2: Implementierung

Leiten Sie die Klasse `net.gumbix.dynpro.IntDynPro` ab und überschreiben Sie die Methoden `n()`, `m()`, `decisions()`, `value()`, `prevStates()` und ggf. `extremeFunction()`, `formatter()` und `rowLabels()`. Schreiben Sie eine `main`-Methode, so dass Sie Ihren Algorithmus ausführen können. Geben Sie die Entscheidungen und die Gesamtkosten aus.



Hochschule
Mannheim/
Fakultät für
Informatik

BIM-6IMB Übungsblatt 1

Prof. Dr. Markus Gumbel
m.gumbel@hs-mannheim.de

Hinweis: Die Bibliothek hierzu finden Sie im Moodle-Kurs unter *Unterlagen für Übungen/Programmbibliothek Dynamische Programmierung*. Bitte beachten Sie die Hinweise dort.

Aufgabe 2.3: Variationen

Angenommen, die wöchentliche Aufbewahrung wäre für $C_s = 1\text{€}$ bzw. $C_s = 2\text{€}$ möglich, da das Labor eine einfache Möglichkeit dazu gefunden hat (z.B. nur Stromkosten für den Kühlschrank oder Inkubator). Wie sähen dann die optimalen Bestellungen und Kosten aus?

Aufgabe 3: Greedy-Superstring

Aufgabe 3.1: Implementierung

Implementieren Sie (in Java, C++, Scala oder Matlab) den in der Vorlesung besprochenen Greedy-Superstring-Algorithmus. Er soll eine Laufzeit von höchstens $O(n^3)$ haben.

Aufgabe 3.2: Unbekannter Text

Wenden Sie Ihren Algorithmus auf die Textfragmente an, die unter *Unterlagen für Übungen/Text-Fragmente* zu finden sind. Diese Datei enthält pro Zeile ein Fragment. Geben Sie den rekonstruierten Text aus.

Aufgabe 3.3: Unbekannte DNA-Sequenz, Teil 1

Wenden Sie Ihren Algorithmus auf die DNA-Fragmente an, die unter *Unterlagen für Übungen/ DNA-Fragmente 1* zu finden sind. Diese Datei enthält pro Zeile ein DNA-Fragment. Geben Sie, soweit möglich, die rekonstruierte DNA-Basenfolge aus. Welches Problem tritt auf?

Aufgabe 3.4: Unbekannte DNA-Sequenz, Teil 2

Wenden Sie Ihren Algorithmus auf die DNA-Fragmente an, die unter *Unterlagen für Übungen/ DNA-Fragmente 2* zu finden sind. Diese Datei enthält die DNA-Fragmente wie in Teil 1 sowie noch weitere. Geben Sie, soweit möglich, die rekonstruierte DNA-Basenfolge aus.

Aufgabe 3.5: Unbekannte DNA-Sequenz, Teil 3

Wenden Sie Ihren Algorithmus auf die DNA-Fragmente an, die unter *Unterlagen für Übungen/ DNA-Fragmente 3* zu finden sind. Diese Datei enthält die gleichen DNA-Fragmente wie in Teil 2 sowie noch weitere. Geben Sie, soweit möglich, die rekonstruierte DNA-Basenfolge aus. Was beobachten Sie?