Ludwig-Maximilians-Universität München
Institut für Informatik
Prof. Dr. Volker Heun

Sommersemester 2019
Semestralklausur
02. August 2019

Algorithmische Bioinformatik I

Vorname	Name	Matrikelnummer
Reihe	**Platz**	**Unterschrift**

Hiermit stimme ich einer Veröffentlichung meines Klausurergebnisses dieser Semestralklausur unter Verwendung meiner Matrikelnummer im Internet zu. *Ja* □ *Nein* □

(Unterschrift)

Allgemeine Hinweise zur Semestralklausur

- Vor der Prüfung ist diese Seite mit Vornamen, Namen, Matrikelnummer, Reihe und Platz **leserlich mit Druckbuchstaben** zu versehen und zu unterschreiben.
- Bitte **nicht** in *roter* oder *grüner* Farbe bzw. **nicht** mit Bleistift schreiben.
- Der Studentenausweis und ein amtlicher Lichtbildausweis sind bereit zu halten.
- Die reine Bearbeitungszeit beträgt **120 Minuten**.
- Es sind insgesamt 40 Punkte zu erreichen, zum Bestehen sind 17 Punkte nötig.

Viel Erfolg!

Hörsaal verlassen von .... bis .... von .... bis ....

Vorzeitig abgegeben um ....

	Hz	A1	A2	A3	A4	A5	$\sum$
Erstkorrektur							
Nachkorrektur							
Zweitprüfer							

Vorname: ______________ Name: ______________ Matrikelnummer: ____________

Aufgabe 1 (8 Punkte)

Gib das Master-Theorem aus der **Vorlesung** an. Spezifiziere hierzu insbesondere die drei verschiedenen Fälle und gib an, welche Lösung der jeweilige Fall besitzt.

Bestimme die Asymptotik von $T(n)$ mithilfe des Master-Theorems aus der **Vorlesung** unter Angabe einer der drei Fälle (siehe oben) mit Begründung bzw. begründe, warum das Master-Theorem nicht anwendbar ist. Es gilt dabei immer $T(1) = 1$:

a) $T(n) = 4 \cdot T(n/2) + n \log(n)$.

b) $T(n) = 3 \cdot T(n/2) + n^2 \log(n)$.

c) $T(n) = 2 \cdot T(n/2) + n \log(n)$.

Lösungsskizze

Seien $a, b, d \in \mathbb{N}$ mit $b > 1$, sei $f(n)$ eine Funktion und sei $\mathcal{C}(n)$ definiert durch die Rekursionsgleichung $\mathcal{C}(n) = a \cdot \mathcal{C}(n/b) + f(n)$ für $n > 1$ und $\mathcal{C}(1) = d$. Dann gilt:

$$\mathcal{C}(n) = \begin{cases} \Theta(n^{\log_b(a)}) & \text{falls } f(n) = O(n^{\log_b(a)-e}) \text{ für ein konstantes } e > 0 \\ \Theta(n^{\log_b(a)} \log(n)) & \text{falls } f(n) = \Theta(n^{\log_b(a)}) \\ \Theta(f(n)) & \text{falls } f(n) = \Omega(n^{\log_b(a)+e}) \text{ für ein konstantes } e > 0 \\ & \text{und } a \cdot f(n/b) \leq c \cdot f(n) \text{ für ein konstantes } c < 1 \end{cases}$$

a) Für das Master-Theorem erhalten wir $a = 4$, $b = 2$ und $f(n) = n \log(n)$. Es gilt $\log_2(4) = 2$ und somit $f(n) = n \log(n) = O(n^{\log_2(4)-e}) = O(n^{\log_b(a)-e})$ für ein geeignetes $e \in (0, 1/2)$. Somit gilt der erste Fall des Master-Theorems und es ist

$$T(n) = \Theta(n^b \log(a)) = \Theta(n^{\log_2(4)}) = \Theta(n^2).$$

b) Für das Master-Theorem erhalten wir $a = 3$, $b = 2$ und $f(n) = n^2 \log(n)$. Es gilt $\log_2(3) \in (1, 2)$ und somit $f(n) = n^2 \log(n) = \Omega(n^{\log_2(3)+e}) = \Omega(n^{\log_b(a)+e})$ für ein geeignetes $e \in (0, 1 - \log_2(3)) \neq \emptyset$. Weiter ist

$$a \cdot f(n/b) = 3 \cdot \left(\frac{n}{2}\right)^2 \cdot \log\left(\frac{n}{2}\right) \leq \frac{3}{4} \cdot n^2 \log(n) \overset{!}{\leq} c \cdot f(n)$$

und somit gilt der dritte Fall des Master-Theorems mit $c = 3/4 < 1$. Damit gilt

$$T(n) = \Theta(f(n)) = \Theta(n^2 \log(n)).$$

c) Für das Master-Theorem erhalten wir $a = 2$, $b = 2$ und $f(n) = n \log(n)$. Somit ist $\log_b(a) = 1$ sowie $f(n) = \omega(n)$ und es ist weder

$$f(n) = n \log(n) = O(n^{1-e}) = O(n^{\log_b(a)-e})$$

für alle $e > 0$ noch

$$f(n) = n \log(n) = \Theta(n) = \Theta(n^{\log_b(a)})$$

noch

$$f(n) = n \log(n) = \Omega(n^{1+e}) = O(n^{\log_b(a)+e})$$

für alle $e > 0$, da $\log(n) = o(n^e)$ für alle $e > 0$.

Aufgabe 2 (8 Punkte)

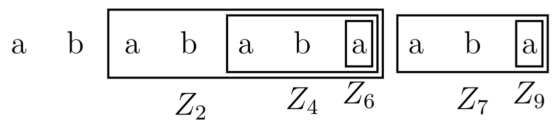
Betrachte das Wort $s = s_0 \cdots s_8 = abababaaba$ und die angegebenen Z-Werte:

i	0	1	2	3	4	5	6	7	8	9
s_i	a	b	a	b	a	b	a	a	b	a
Z_i		0	5	0	3	0	?	?	?	1

a) Zeichne die Z-Boxen für die Z-Werte $Z_1, \ldots, Z_9$ ein.

b) Gib die Berechnung der drei fehlenden Z-Werte Z_6, Z_7 und Z_8 gemäß dem in der Vorlesung angegebenen Linearzeit-Algorithmus an und begründe (bzgl. der einzelnen Fälle), warum der Algorithmus die einzelnen Schritte ausführt.

Lösungsskizze



Die Z-Boxen Z_1, Z_3, Z_5 und Z_8 sind leer und daher nicht eingezeichnet.

Berechnung von Z_6: Für die Berechnung von Z_6 ist die rechteste bislang gefundene Z-Box Z_2, somit ist $r = 6$ und $\ell = 2$. Für $k = 6$ gilt also $k = 6 \leq 6 = r$, d.h. Fall 2. Wir betrachten die Z-Box $Z_{k-\ell} = Z_4 = 3$. Diese ist nicht kürzer als die aktuell betrachtete Bereich der rechtesten Z-Box ($Z_{k-\ell} = Z_4 = 3 \geq 1 = 6 - 6 + 1 = r - k + 1$). Somit ist $Z_6 \geq r - k + 1 = 6 - 6 + 1 = 1$ und der aktuelle Wert muss mit Vergleichen ermittelt werden (Fall 2b). Hierfür sind kein weiteren erfolgreicher Zeichenvergleiche möglich und somit ist $Z_6 = 1$ nun die rechteste Z-Box bleibt mit $r = 6$ und $\ell = 2$.

Berechnung von Z_7: Für die Berechnung von Z_6 ist die rechteste bislang gefundene Z-Box Z_2, somit ist $r = 6$ und $\ell = 2$. Für $k = 7$ gilt also $k = 7 > 6 = r$, d.h. Fall 1. Wir vergleichen also Zeichenweise ab Position 7 mit dem Präfix von s. Dies liefert $Z_7 = 3$ mit $\ell_7 = 7$ und $r_7 = 9$.

Berechnung von Z_8: Für die Berechnung von Z_8 ist die rechteste bislang gefundene Z-Box Z_7. Somit ist $r = 9$ und $\ell = 7$. Für $k = 8$ gilt also $k = 8 < 9 = r$, d.h. Fall 2. Wir betrachten die Z-Box $Z_{k-\ell} = Z_1 = 0$. Diese ist kürzer als der aktuell betrachtete Bereich der rechtesten Z-Box ($Z_{k-\ell} = Z_1 = 0 < 2 = 9 - 8 + 1 = r - k + 1$), wir befinden uns also im Fall 2a. Somit ist $Z_8 = Z_1 = 0$.

Vorname: ______________ Name: ______________ Matrikelnummer: ______________

Aufgabe 3 (8 Punkte)

Löse die folgende Rekursionsgleichung **mit Hilfe der allgemeinen Lösung für lineare Rekursionsgleichungen**:

$$f_n = 2 \cdot f_{n-1} + 1 \quad \text{für } n \geq 1, \qquad \text{und} \qquad f_0 = 0.$$

Lösungsskizze

Betrachte die Rekursionsgleichung für n und $n-1$:

$$\begin{aligned} f_n &= 2 \cdot f_{n-1} + 1, \\ f_{n-1} &= 2 \cdot f_{n-2} + 1. \end{aligned}$$

Ziehen wir die zweite von der ersten Gleichung ab, so erhalten wir eine homogene lineare Rekursionsgleichung

$$f_n = 3 \cdot f_{n-1} - 2 \cdot f_{n-2}$$

mit den Anfangsbedingungen $f_0 = 0$ und $f_1 = 2 \cdot f_0 + 1 = 1$.

Wir betrachten nun das charakteristische Polynom $\chi(n)$ für diese Rekursionsgleichung:

$$\chi(n) = n^2 - 3n + 2.$$

Die Nullstellen ergeben sich (beispielsweise mit der p, q-Formel) zu

$$x_{1,2} = \frac{3}{2} \pm \sqrt{\frac{9}{4} - 2} = \frac{3}{2} \pm \sqrt{\frac{1}{4}}.$$

Also sind 1 und 2 die beiden Nullstelle des charakteristischen Polynoms. Somit hat die Lösung der Rekursionsgleichung die Form:

$$f_n = a \cdot (1)^n + b \cdot (2)^n = a + b \cdot 2^n.$$

Mit den Anfangsbedingungen ergibt sich für a und b:

$$\begin{aligned} 0 &= f_0 = a + b, \\ 1 &= f_1 = a + 2 \cdot b. \end{aligned}$$

Subtraktion der ersten von der zweiten Gleichung liefert $b = 1$. Somit ist $a = -1$, also ist $f_n = 2^n - 1$.

Vorname: ______________ Name: ______________ Matrikelnummer: ____________

Aufgabe 4 (8 Punkte)

Beweise $\Delta\binom{m+x}{x} = \binom{m+x}{x+1}$ für $m \geq 0$ und berechne damit sowie mithilfe der diskreten Stammfunktion und partieller Integration für $m \in \mathbb{N}_0$:

$$\sum_{i=0}^{n-1} i \cdot \binom{m+i}{i+1}$$

Hinweis: Es gilt $\binom{m}{n} = \frac{m\cdots(m-n+1)}{n\cdots 1}$.

Lösungsskizze

Es gilt nach Definition:

$$\begin{aligned}
\Delta\binom{m+x}{x} &= \binom{m+x+1}{x+1} - \binom{m+x}{x} \\
&= \frac{(m+x+1)\cdots(m+1)}{(x+1)\cdots 1} - \frac{(m+x)\cdots(m+1)}{x\cdots 1} \\
&= \frac{(m+x+1)\cdots(m+1)}{(x+1)\cdots 1} - \frac{x+1}{x+1} \cdot \frac{(m+x)\cdots(m+1)}{x\cdots 1} \\
&= \frac{(m+x)\cdots(m+1)\cdot[(m+x+1)-(x+1)]}{(x+1)\cdots 1} \\
&= \frac{(m+x)\cdots(m+1)\cdot m}{(x+1)\cdots 1} \\
&= \binom{m+x}{x+1}.
\end{aligned}$$

Nach der partiellen Integration gilt $\sum(f \cdot \Delta g) = fg - \sum(\Delta f)(Eg)$ mit $f(x) = x$ und $g(x) = \binom{m+x}{x}$ und daher $(\Delta g)(x) = \binom{x}{m-1}$ sowie $(\Delta f)(x) = 1$:

$$\begin{aligned}
\sum_{i=0}^{n-1} i \cdot \binom{m+i}{i+1} &= \left[i \cdot \binom{m+i}{i}\right]_0^n - \sum_{i=0}^{n-1} 1 \cdot \binom{m+1+i}{i+1} \\
&\quad \text{es gilt auch } \Delta\binom{m+1+x}{x} = \binom{m+1+x}{x+1} \\
&= \left[i \cdot \binom{m+i}{i}\right]_0^n - \left[\binom{m+1+i}{i}\right]_0^n \\
&= n \cdot \binom{m+n}{n} - 0 - \left[\binom{m+n+1}{n} - \binom{m+1}{0}\right] \\
&= n \cdot \binom{m+n}{n} - \binom{m+n+1}{n} + 1.
\end{aligned}$$

Vorname: ______________ Name: ______________ Matrikelnummer: ____________

Aufgabe 5 (8 Punkte)

Für ein Wort $w = w_1 \cdots w_m \in \Sigma^m$ bezeichnet $w^R = w_m \cdots w_1 \in \Sigma^m$ das *gespiegelte Wort* zu w.

Konstruiere einen Linearzeit-Algorithmus, der für ein Wort $t \in \Sigma^*$ ein kürzestes Teilwort w von t findet, dass nur genau einmal in t vorkommt und dessen gespiegeltes Wort w^R nicht in t vorkommt.

Hinweis: Korrektheitsbeweis und Laufzeitanalyse nicht vergessen!

Lösungsskizze

Wir konstruieren nun für $t\sharp t^R\$$ mit $\sharp \neq \$ \notin \Sigma$ den zugehörigen Suffix-Baum T. Die Größe des Suffix-Baumes beträgt $O(2n) = O(n)$. Mit einer Tiefensuche entfernen wir alle Teilbäume, die über das Zeichen $\sharp$ erreichbar sind und zwar nach dem Zeichen $\sharp$. Somit enden nun alle Pfade im Suffix-Baum mit dem Zeichen $\sharp$ oder \$.

Mit einer weiteren Tiefensuche markieren wir alle Knoten des Baumes, die in ihrem Teilbaum nur noch ein $\sharp$ enthalten oder ein Blatt sind und das letzte Zeichen den Kantenlabels $\sharp$ ist. Auf den Kanten zu diesen Knoten enden nun Teilwörter von t, die genau einmal in t auftreten. Nun markieren wir mit einer weiteren Tiefensuche alle Knoten, von denen aus kein $\$ erreichbar ist$ und die Blätter deren Kantenlabel nicht mit $\$ endet$. Auf den Kanten zu diesen Knoten enden keine Teilwörter von t^R, die genau einmal in t auftreten.

Auf allen so markierten Knoten ist der String, der zum Eltern-Konten gehort plus dem ersten Zeichen auf der Kante ein Kandidat für ein kürzeste Wort w, das genau einmal in t und nicht in t^R auftritt. Dieses Wort w ist also genau einmal teilwort von t und w^R taucht nicht in t auf. Unter all diesen Kandidaten müssen wir nur ein kürzestes Wort finde und ausgeben.

Da sowohl die Konstruktion des Suffix-Baumes in Zeit $O(2n) = O(n)$ als auch die drei bis vier Tiefensuchen nur jeweils $O(n)$ Zeit benötigen, ist die Gesamtaufzeit $O(n)$.