Übung 7
Algorithmische Bioinformatik

WS 15/16

Yvonne Mayer

Lösungen/ Wettbewerb Übung 6

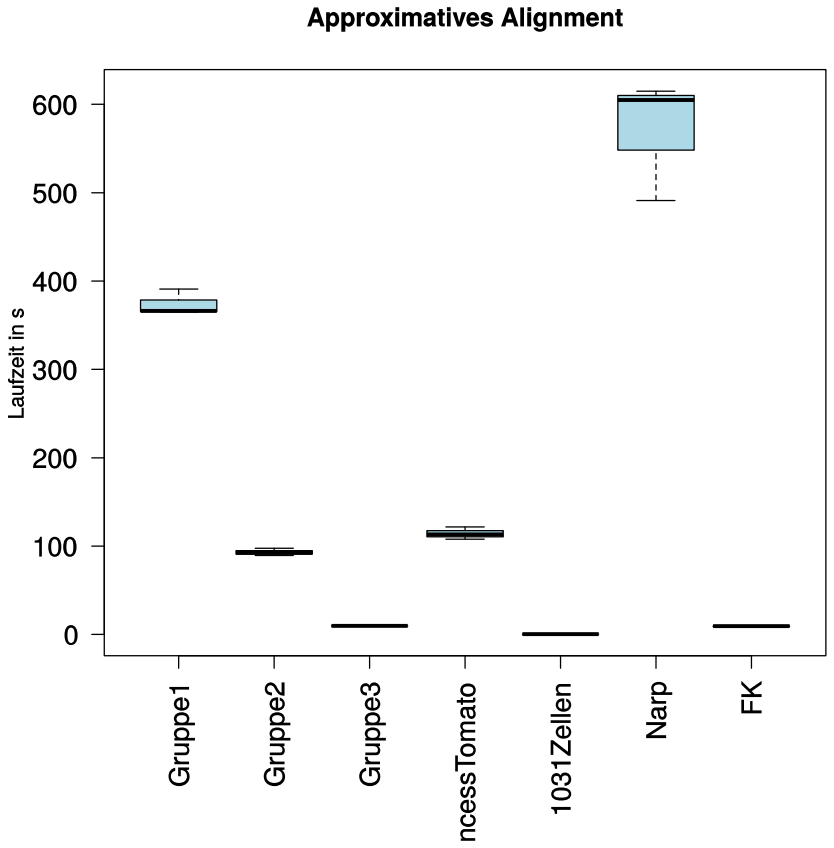
Lösungen Übung 6 vorstellen

- Approximatives String-Matching (20P)

GruppeAB ?

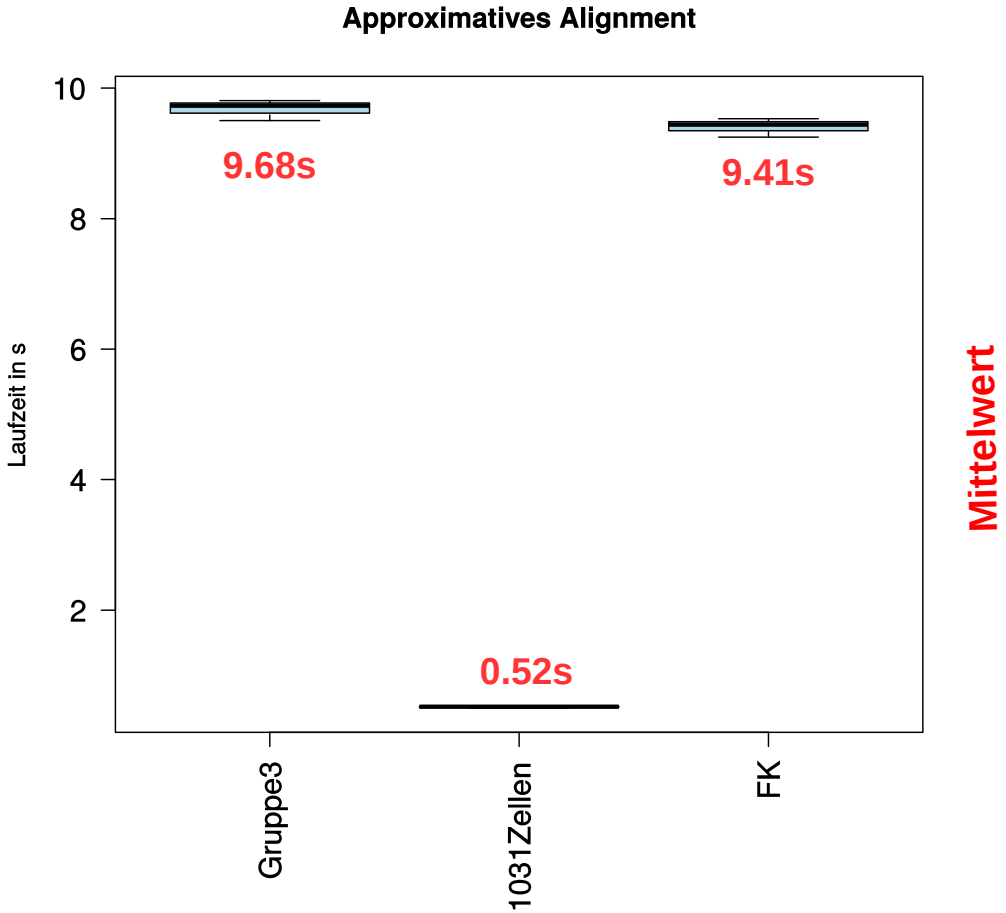
Wettbewerb Übung 6: Approximatives String-Matching

Ergebnisse aller Gruppen mit richtiger Lösung:



Wettbewerb Übung 6: Approximatives String-Matching

Ergebnisse aller Gruppen mit Lösung unter 10sec:



Wettbewerb

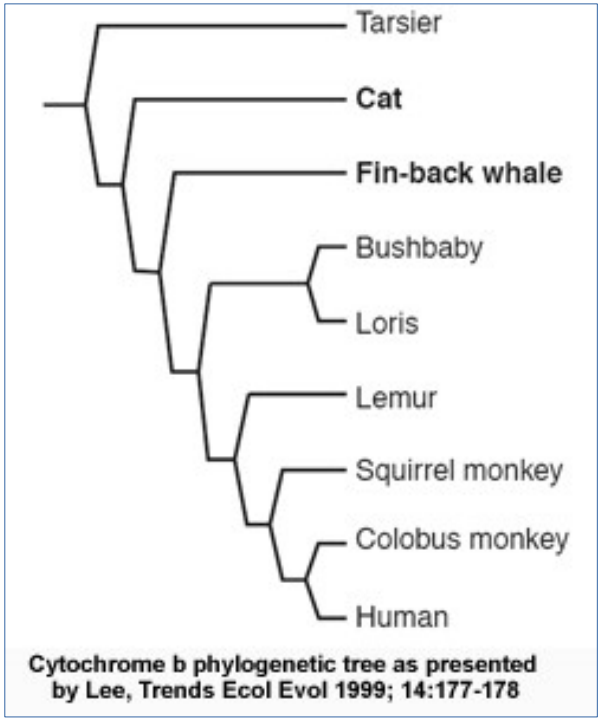
	Gruppe0	Gruppe1	Gruppe2	Gruppe3	PrincessTomato	1031Zellen	LL	Narp	FK
Challenge1 (beliebiges Stringmatching)	0	0	0	0	0	3	0	1	2
Challenge2 (Boyer Moore)	0	0	1	1	0	3	0	0	2
Challenge3 (q-grams)	0	0	0	2	0	1	0	0	3
Challenge4 (Suffix-arrays)	0	0	0	0	0	3	0	2	1
Challenge5 (Globales Alignment)	0	1	0	0	0	2	0	1	3
Challenge6 (ApprString -Maching)	0	0	1	2	0	3	0	0	2
Summe	0	1	2	5	0	15	0	5	13

(Ergebnisse Challenge 6 unter Vorbehalt, bis Korrektur vollständig abgeschlossen)

Übung 7
(Neighbor Joining)

Neighbor Joining

- **Gegeben:** Ähnlichkeiten zwischen mehreren DNA-Sequenzen
- **Gesucht:** Ein phylogenetischer Baum, welcher die Verwandtschaft als „Stammbaum“ darstellt



Cytochrome b phylogenetic tree as presented by Lee, Trends Ecol Evol 1999; 14:177-178

Algorithmus (1)

(1) Berechnung der paarweisen Ähnlichkeit (mittels Alignment) aller Sequenzen.

(2) Berechnung der durchschnittlichen Distanz u_i zu jedem Taxon (DNA-Sequenz).

$$u_i = \sum_{k \neq i} \frac{D[i,k]}{n-2}$$

(3) Suche das Clusterpaar (i,j) mit dem kleinsten Abstand

$$D[i,j] - u_i - u_j = min$$

Algorithmus (2)

(4) Erzeuge einen neuen Clusterknoten (i,j) und berechne die Distanz zwischen neuem Taxa und den „restlichen“ Taxa.

$$D[k,ij] = \frac{D[i,k] + D[k,j] - D[i,j]}{2}$$

(5) Lösche die beiden Taxa i und j.

(6) Iteriere bis nur noch ein Taxa existiert.

Beispiel

Schritt (2)

	A	B	C	D
A		6	8	12
B			4	8
C				6
u_i	**13**	9	9	13

$$u_i = \sum_{k \neq i} \frac{D[i,k]}{n-2}$$

$$u_i = \frac{(6+8+12)}{2} = 13$$

Schritt (3)

	A	B	C	D
A		-16	-14	-14
B			-14	-14
C				**-16**

$$D[i,j] - u_i - u_j = min$$

$$(6-9-13)$$

Restlichen Schritte

	A	B	CD
A		6	**7**
B			3

$$\frac{(8+12-6)}{2}$$

Beispiel ...

Schritt (2)

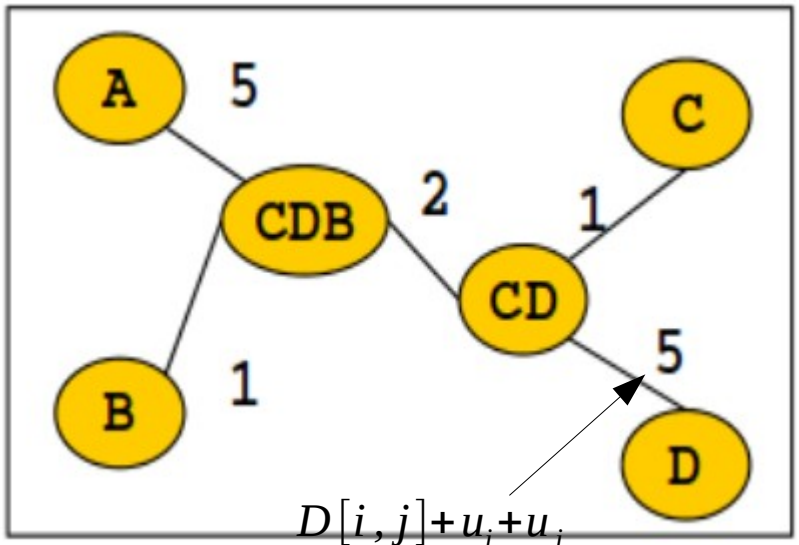
	A	B	CD
A		6	7
B			3
u_i	13	9	10

Schritt (3)

	A	B	CD
A		-16	-16
B			-16

Restlichen Schritte

	A	CDB
A		5



$$v_{i,u} = \frac{D[i,j] + u_i + u_j}{2}$$

Übung 7.1

- Implementieren Sie den Neighbor Joining Algorithmus um den phylogenetischen Baum zu berechnen (20P).

- **Ausgangspunkt:**
  - 2000 DNA Sequenzen von *Escherichia coli*.
  - Die paarweise Ähnlichkeit zwischen allen Sequenzen wurde bereits vorberechnet.

- **Aufgabe:**
  - Einlesen der paarweisen Ähnlichkeitsmatrix.
  - Berechnung des phylogenetischen Baums als Reihe von Cluster-Merge Schritten.
  - Beispiel: Wenn als erstes die Sequenzen 1 und 4 zu 14 gemergt werden, dann 5 und 7 zu 57, dann 14 und 3 zu 143, dann soll die Ausgabe so aussehen: (1,4), (5,7),(14,3).
  - Bitte mit Kommentaren im Sourcecode (3P) und einer kurzen Erläuterung in einem PDF was getan wurde (3P).

Inputformat

- Input ist eine Ähnlichkeitsmatrix.
- Die Ähnlichkeitsmatrix ist vorberechnet und symmetrisch.
- Matrix für unser Beispiel:

```
99;6;8;12
6;99;4;8
8; 4;99;6
12;8;6;99
```

Outputformat

- Ausgabe der Clusterschritte auf STDOUT.
- Ausgabe für das Beispiel:

```
(3,4)
(34,2)
(342,1)
```

Programmaufruf

- Inputfile ist unkomprimiert
- Neighbor-Joining muss wie folgt aufrufbar sein:

  ```
  java jar Assignment67GrXY.jar f1
  f1 ... paarweise Ähnlichkeitsmatrix
  ```

  Ausgabe auf STDOUT

Wettbewerb

- Wir vergeben Punkte für die schnellste Berechnung des phylogenetischen Baums.
- Wir verwenden eine andere Ähnlichkeitsmatrix ähnlicher Größenordnung. Das Programm soll mit Matrizen von 2x2 bis 2000x2000 umgehen können.
- Wir messen die Gesamtlaufzeit Ihres Verfahrens für diese Übung (mittels Linux „`time`“, `user+sys`), Parallelisierung lohnt nicht.
- Wettbewerbspunkte:
  - Platz 1: 3 Punkte
  - Platz 2: 2 Punkte
  - Platz 3: 1 Punkt

Abgabe

- Abgabe bis Sonntag den 07.02.2016 um 23:59 Uhr

- Abgabe per Email an: mayeryvo@informatik.hu-berlin.de
  (gerne auch Fragen zur Übung per Email,
  Code Anfragen alte Aufgaben!)

- Jar Datei mit Quellcode (kommentiert!)
  (Dateiname: Assignment7_GrXY.jar, jar ggf. als rar verpacken)

- Kompilierung unter Java 1.7, Jar Datei auf gruenau2 testen,
  Abgaben ohne Quellcode werden ignoriert!

- Bitte die geschätzte Bearbeitungszeit mitteilen
  (z.B. 10h für zwei Teilnehmer)