

Universität Leipzig Institut für Informatik Bioinformatik/IZBI	Algorithmen und Datenstrukturen II SoSe 2024 – Freiwillige Serie 7		
P.F. Stadler, T. Gatter	Ausgabe am 14.05.2024	Lösung am 21.05.2024	Seite 1/2

Algorithmen und Datenstrukturen II

SoSe 2024 – Serie 7

1 Editierdistanz

Gegeben sind die folgenden zwei Zeichenketten/Wörter:

1: cultural

2: kulturell

Berechnen Sie deren Editierdistanz mit dem in der Vorlesung vorgestellten DP Algorithmus. Geben Sie sowohl die vollständige Matrix D als auch **alle** optimalen Alignments an. Benutzen Sie ein Einheitskostenmodell in dem jede Editieroperation – egal ob Insertion, Deletion, oder Substitution – gleich viel kostet.

Lösung:

	-	k	u	l	t	u	r	e	l	l
-	0	1	2	3	4	5	6	7	8	9
c	1	1	2	3	4	5	6	7	8	9
u	2	2	1	2	3	4	5	6	7	8
l	3	3	2	1	2	3	4	5	6	7
t	4	4	3	2	1	2	3	4	5	6
u	5	5	4	3	2	1	2	3	4	5
r	6	6	5	4	3	2	1	2	3	4
a	7	7	6	5	4	3	2	2	3	4
l	8	8	7	6	5	4	3	3	2	3

Distanz: 3

Optimale Alignments:

cultur-al
kulturell

cultura-l
kulturell

cultural-
kulturell

2 Optimale Matrixmultiplikation

Gegeben seien die folgende 4 Matrizen und ihre Dimensionen:

- $A_1 : 10 \times 5$
- $A_2 : 5 \times 4$

Universität Leipzig Institut für Informatik Bioinformatik/IZBI	Algorithmen und Datenstrukturen II SoSe 2024 – Freiwillige Serie 7		
P.F. Stadler, T. Gatter	Ausgabe am 14.05.2024	Lösung am 21.05.2024	Seite 2/2

- $A_3 : 4 \times 2$
- $A_4 : 2 \times 6$

Bestimmen Sie für den Term $A_1 A_2 A_3 A_4$ eine optimale Klammerung, die eine minimale Anzahl von Rechenoperationen erfordert. Verwenden Sie dazu den in der Vorlesung vorgestellten Dynamic Programming Algorithmus. Geben Sie sowohl die Matrix als auch die optimale Klammerung an.

Lösung:

$$r_1 = 10, r_2 = 5, r_3 = 4, r_4 = 2, r_5 = 6$$

	10 A_1 5	5 A_2 4	4 A_3 2	2 A_4 6
10 A_1 5	0 $[A_1]$	200 $[A_1 A_2]$	140 $[A_1 [A_2 A_3]]$	260 $[[A_1 [A_2 A_3]] A_4]$
5 A_2 4		0 $[A_2]$	40 $[A_2 A_3]$	100 $[[A_2 A_3] A_4]$
4 A_3 2			0 $[A_3]$	48 $[A_3 A_4]$
2 A_4 6				0 $[A_4]$

$$C_{1,2} = C_{1,1} + C_{2,2} + r_1 r_2 r_3 = 0 + 0 + 10 \cdot 5 \cdot 4 = 200$$

$$C_{2,3} = C_{2,2} + C_{3,3} + r_2 r_3 r_4 = 0 + 0 + 5 \cdot 4 \cdot 2 = 40$$

$$C_{3,4} = C_{3,4} + C_{4,4} + r_3 r_4 r_5 = 0 + 0 + 4 \cdot 2 \cdot 6 = 48$$

$$C_{1,3} = \min($$

$$C_{1,1} + C_{2,3} + r_1 r_2 r_4 = 0 + 40 + 10 \cdot 5 \cdot 2 = 140,$$

$$C_{1,2} + C_{3,3} + r_1 r_3 r_4 = 200 + 0 + 10 \cdot 4 \cdot 2 = 280$$

$$) = 140$$

$$C_{2,4} = \min($$

$$C_{2,2} + C_{3,4} + r_2 r_3 r_5 = 0 + 48 + 5 \cdot 4 \cdot 6 = 168,$$

$$C_{2,3} + C_{4,4} + r_2 r_4 r_5 = 40 + 0 + 5 \cdot 2 \cdot 6 = 100$$

$$) = 100$$

$$C_{1,4} = \min($$

$$C_{1,1} + C_{2,4} + r_1 r_2 r_5 = 0 + 100 + 10 \cdot 5 \cdot 6 = 400,$$

$$C_{1,2} + C_{3,4} + r_1 r_3 r_5 = 200 + 48 + 10 \cdot 4 \cdot 6 = 488,$$

$$C_{1,3} + C_{4,4} + r_1 r_4 r_5 = 140 + 0 + 10 \cdot 2 \cdot 6 = 260$$

$$) = 260$$

Die optimale Klammerung ist somit $[[A_1 [A_2 A_3]] A_4]$ mit 260 Rechenoperationen.