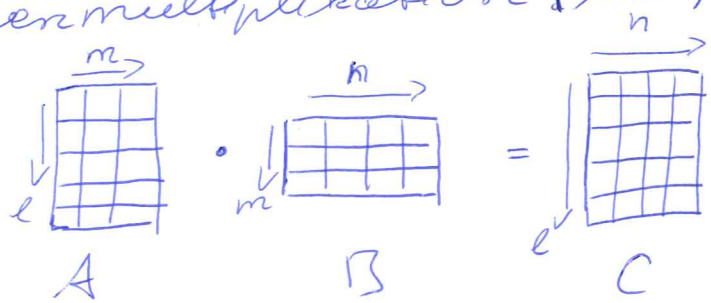
- große Übung AuD2 10.6.2015
- letzte gr. Übung:
  - LCS mit DP und D&C

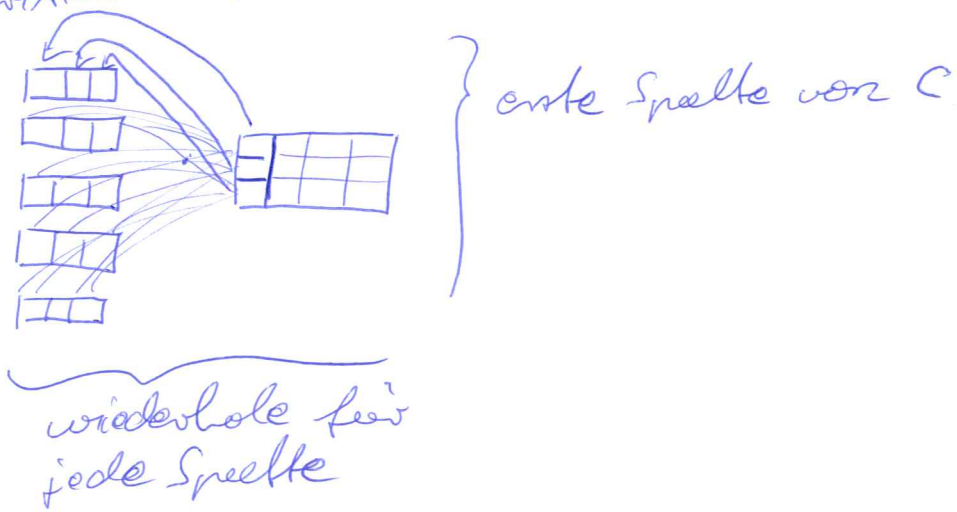
    $\downarrow$ $O(n \cdot m)$ $\qquad$ $\downarrow$ $O(2^{\min\{n,m\}}) \xrightarrow{\text{Memoisation}} O(n \cdot m)$

  - Memoisation
    - erinnere dich -> berechne Eintrag ~~LCS[i,j]~~ LCS(i,j) nur einmal
- Verallg.:
  - sei d Dim., $OPT(\underbrace{0,\dots,0}_{d})$ sei DP
  - berechne $OPT(i_1,\dots,i_d)^d$ $\forall\, i_1,\dots,i_d$ nur einmal
- Heute:
  - Verallg. von LCS: globales Alignment
    + Umsetzung: Iterativ DP und D&C mit Mem.
  - Hinweise zu Blatt 4 Aufg 1

Matrizenmultiplikation (MM)



- Schrittweise:



- Bsp

$$\begin{pmatrix} 5 & 3 & 2 \\ 7 & 1 & 2 \\ 1 & 3 & 4 \\ 2 & 5 & 6 \\ 4 & 2 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 2 & 3 \\ 0 & 2 & 1 & 5 \\ 1 & 4 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 7 & 14 & 13 & 32 \\ 9 & 10 & 15 & 28 \\ 5 & 22 & 5 & 22 \\ 8 & 34 & 9 & 37 \\ 5 & 12 & 10 & 23 \end{pmatrix}$$

- Allgemeiner: $M[i][j]$ Eintrag i-te Zeile, j-te Spalte

$$C[i][j] = \sum_{k=1}^{m} A[i][k] \cdot B[k][j]$$

(*)

- MM ist assoziativ:

  M_1, M_2 und M_3 Matrizen mit ent. Dimensionen:

  $$(M_1 \cdot M_2) \cdot M_3 = M_1 \cdot (M_2 \cdot M_3)$$

(*) - # ben. Multiplikationen für gesamt C: $\underbrace{l}_{\#\text{Zeilen}} \cdot \underbrace{n}_{\#\text{Spalten}} \cdot \underbrace{m}_{\#\text{ben. für einen Eintrag von C}}$

- Konsequenz:
  - Sei $\langle M_1, \dots, M_n \rangle$ Folge von Matrizen mit $M_i \in \mathbb{R}^{d_i} \times \mathbb{R}^{d_{i+1}}$ für $i \in \{1, \dots, n-1\}$
    (#Spalten = # Zeilen in folg. Matrix)

  $\Rightarrow$ bel. Klammerung (Tonwelle) definiert

  - Hinweis $A \cdot B \cdot C \overset{1}{=} (A \cdot B) \cdot C$
    bzgl. Reihenfolge

  - Also: Frage nach Klammerung $\overset{1}{=}$ rek. Frage nach "Splitten" (bin)

$$(M_1 \cdot M_2 \cdot M_3) \cdot (M_4 \cdot M_5 \cdot M_6)$$

$$(M_1 \cdot M_2) \cdot M_3 \qquad\qquad M_4 \cdot (M_5 \cdot M_6)$$

$$(M_1) \cdot (M_2) \qquad\qquad (M_5) \cdot (M_6)$$

$\Rightarrow$ wo splitten? $\overset{1}{=}$ Entscheidung der Klammerung

$\underbrace{\qquad\qquad\qquad\qquad}_{\text{auf jeder Ebene}}$

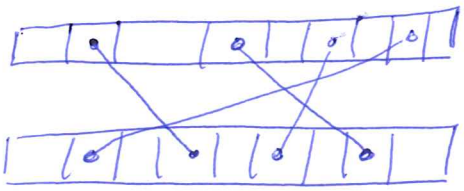
- Globales Alignment
  → wie aufwendig Wörter ausrichten
- Bsp.: BRÜCKE & BÜCHSE

  BRÜCK__E
  B_ÜC_HSE } Ausrichtung

  ⇒ 4 Lücken + 0 Mismatches
- Formal
  - $M \subseteq \{x_1, \dots, x_n\} \times \{y_1, \dots, y_m\}$ <u>Matching</u>

    $:\Longleftrightarrow$

    1.) $\forall i \in \{1, \dots, n\}: |\{(x_i, x_d) \in M\}| \leq 1$ und
    2.) $\forall j \in \{1, \dots, m\}: |\{(x_d, x_j) \in M\}| \leq 1$

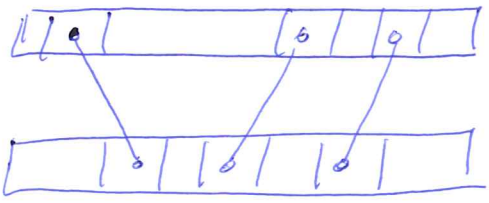


  - Matching M heißt Zuordnung

    $:\Longleftrightarrow$

    für $(x_i, y_j) \in M$, $(x_d, y_e) \in M$ gilt

    $\underbrace{[x_i < x_d \iff y_j < y_e]}_{\text{keine Kreuzungen}}$



  - Kosten der Zuordnung M:
    1. pro nicht gematchtem Buchstabe aus X oder Y:
       penalty $\delta > 0$
    2. Mismatch-Kosten ϑ_{x_i, y_i}

       (i.A. $\vartheta_{pp} = 0$)

       zum ~~Beispiel~~ Beispiel: $\delta = 2$, $\vartheta_{pp'} = 3$ wenn $p \neq p'$

       $\vartheta_{pp} = 0$

- Dyn. Prog.:

$OpA(i,j)$ = min. Zuordnungs-Kosten für X^i, Y^j

→ Rekursion:

$OpA(i,0) = \delta \cdot i$

$OpA(0,j) = \delta \cdot j$

$$OpA(i,j) = \min\{ \gamma_{X[i]Y[j]} + OpA(i-1,j-1),\ \delta + OpA(i,j-1),\ \delta + OpA(i-1,j)\}$$

- Bsp.: $\delta = 2$, $\gamma_{pp'} = 3$ mit $p \neq p'$

X = BRÜCKE

Y = BÜCHSE

		j	B'	R	Ü	C	K	E
i	0		1	2	3	4	5	6
		0	2	4	6	8	10	12
B	1	2	0	2	4	6	8	10
Ü	2	4	2	3	2	4	6	8
C	3	6	4	5	4	2	4	6
H	4	8	6	7	6	4	5	7
S	5	10	8	9	8	6	7	9
E	6	12	10	11	10	8	9	7

$\Rightarrow OpA(n,m) = OpA(6,6) = 7$

~~Matching~~ OpA Zuordnung:

B R Ü C K E
| | | | | | |
B _ Ü C H S E

- Alg für glob. Align. mit DP:

```
globAlignDP(γ, δ, X, Y)
    n = |X|
    m = |Y|                                   } 0.
    init OPT[n][m]
    for (i=0,...,n){
        OPT[i][0] = δ·i                       } 1.
    }
    for (j=0,...,m){
        OPT[0][j] = δ·j                       } 2.
    }
    for (i=1,...,n){
        for (j=1,...,n){
            A = γ(X[i], Y[j]) + OPT[i-1, j-1]
            B =     δ         + OPT[i, j-1]   } 3.
            C =     δ         + OPT[i-1, j]   } 4.
            if (A ≤ B){min = A} else {min = B}
            if (C ≤ min){min = C}
            OPT[i][j] = min
        }
    }
    return OPT[n][m] } 5.
}
```

- Laufzeit

0.	Initialisierung	$O(1)$ oder $O(n \cdot m)$
1.	Startwerte	$O(n)$
2.	"	$O(m)$
3.	Berechnung von $OPT[i][j]$ Konst.	$O(1)$
4.	geschachtelte Schleifen	$n \cdot m \cdot O(1) = O(n \cdot m)$

- Korrektheit
  - Z: $OPT[n][m]$ ist Wert für glob. Align.
  - Beweis durch vollst. Ind.

- zwei Indizes i, j
- IA: 1.) $OPT[i][0] \hat{=}$ ersten i-Buchstaben nicht zugeordnet $\Rightarrow$ i-mal penalty $\Rightarrow i \cdot \delta = OPT[i][0]$

  2.) $OPT[0][j] = j \cdot \delta$ (analog)

  $\forall i \in \{0, \dots, n\}$, $\forall j \in \{0, \dots, m\}$

- IV: $\forall i \leq n-1, \forall j \leq m-1: OPT[i][j] =$ Wert von opt. glob. Align von $X[0], \dots, X[i]$ und $Y[0], \dots, Y[j]$

- IS: Es gilt: $OPT[n][m]$ kann nur aus $OPT[n][m-1]$, $OPT[n-1][m]$ oder $OPT[n-1][m-1]$ gebildet werden

$$\Rightarrow OPT[n][m] = \min\{ OPT[n][m-1] + \delta,\ OPT[n-1][m] + \delta,\ OPT[n-1][m-1] + \gamma_{X[n]Y[m]} \}$$

$\square$

- Alg. für glob. Align mit D&C und Mem.

```
globAlignDC und Mem(γ, δ, i, j, X, Y, mem)
    if (i == 0){return δ·j}, if (j == 0){return i·δ}
    if (mem[i][j] ≠ ∞){return mem[i][j]}
    else{
        A = globAlignDC und Mem(γ, δ, i-1, j, X, Y, mem) + δ
        B =          "          (γ, δ, i, j-1, X, Y, mem) + δ
        C =          "          (γ, δ, i-1, j-1, X, Y, mem) + γ(X[i], Y[j])

        mem[i][j] = min{A, B, C}
        return mem[i][j]
    }
}
```

- Korrektheit + Laufzeit: z.o.