Matrix Chain Multiplication

Suppose we have matrix A, B

A [ ] * B [ ] = C [ ]

5x4   4x3   5 × 3 × 4 (60)

i.e. To perform multiplication of 2 matrices coloum of first matrice should be same as row of second matrices.

Value 60 implies to multiplication cost.

- Suppose we have A1 · A2 · A3 · A4 matrices to be multiplied. Each of the combination will give different result.

  i.e. (A1·(A2 (A3·A4))), ((A1·A2)·(A3·A4)), ((A1 (A2·A3))A4)
  (A1((A2·A3))·A4)), (((A1·A2)·A3)A4)

Dynamic programming focusses on the concept of finding optimal solution after trying out all the various possibilities. To do so, it forms substructure of problem which can be stored with resultant value to be reused.

A1 (5 × 4)   A2 (4 × 6)   A3 (6 × 2)   A4 (2 × 7).

<u>Step 1:</u> Construct a DP table. (Dynamic programming)

It is a 2 dimensional array matrix used to store the results of subproblems in a bottom-up dynamic programming. It helps avoid redundant computation by storing intermediate results, which can be referenced later instead of recomputing them.

m

(1)

	1	2	3	4
1				
2				
3				
4				

s

	1	2	3	4
1				
2				
3				
4				

Step 2: compute values for single matrices.

A1 $(5\times4) \Rightarrow m[1,1] \rightarrow 0$

A2 $(4\times6) \Rightarrow m[2,2] \rightarrow 0$

A3 $(6\times2) \Rightarrow m[3,3] \rightarrow 0$

A4 $(2\times7) \Rightarrow m[4,4] \rightarrow 0$

m

	1	2	3	4
1	0			
2		0		
3			0	
4				0

Step 3: compute values for Two matrices.

A1 · A2

m[1,2]

$= 5\times4 \cdot 4\times6$

$= 5\times4\times6$

$= 120$

A2 · A3

m[2,3]

$= 4\times6 \cdot 6\times2$

$= 4\times6\times2$

$= 48$

A3 · A4

m[3,4]

$= 6\times2 \cdot 2\times7$

$= 6\times2\times7$

$= 84$

$\Rightarrow m[1,1] + m[2,2] + 5\times4\times6$

$\Rightarrow m[2,2] + m[3,3] + 4\times6\times2$

$\Rightarrow m[3,3] + m[4,4] + 6\times2\times7$

	1	2	3	4
1	0	120		
2		0	48	
3			0	84
4				0

Step 4: Compute values for 3 matrices.

$m[1,3]$

$A1 \cdot (A2 \cdot A3)$

$= 5\times4 \cdot (4\times6 \cdot 6\times2)$

$= m[1,1] + m[2,3] + 5\times4\times2$

$= 0 + 48 + 40$

$= 88$

$(A1 \cdot A2) \cdot A3$

$= (5\times4 \cdot 4\times6) \cdot 6\times2$

$= m[1,2] + m[3,3] + 5\times6\times2$

$= 120 + 0 + 60$

$= 180$

s

	1	2	3	4
1		1		
2			2	
3				3
4				

m

	1	2	3	4
1	0	120	88	
2		0	48	
3			0	84
4				0

$m[2,4]$

$A2 \cdot (A3 \cdot A4)$

$= 4\times6 \cdot (6\times2 \cdot 2\times7)$

$= m[2,2] + m[3,4] + 4\times6\times7$

$= 0 + 84 + 168$

$= 252$

$(A2 \cdot A3) \cdot A4$

$= (4\times6 \cdot 6\times2) \cdot 2\times7$

$= m[2,3] + m[4,4] + 4\times2\times7$

$= 48 + 0 + 56$

$= 104$

m

	1	2	3	4
1	0	120	88	
2		0	48	104
3			0	84
4				0

Step 5: Compute values for 4 matrices.

m[1, 4] =

A1	A2	A3	A4
5×4	4×6	6×2	2×7

$$= \min \{ m[1,1] + m[2,4] + 5 \times 4 \times 7,$$

$$m[1,2] + m[3,4] + 5 \times 6 \times 7,$$

$$m[1,3] + m[4,4] + 5 \times 2 \times 7 \}.$$

equivalent to A·(B·C·D), (A·B)·(C·D), (A·B·C)·D

consider A·(B·C·D)

5×4 (4×6 · 6×2 · 2×7)

(A·B)·(C·D)

(5×4 · 4×6) · (6×2 · 2×7)

(A·B·C)·D

(5×4 4×6 6×2) · 2×7 =

$$= \min \{ 0 + 104 + 140, 120 + 84 + 210, 88 + 0 + 70 \}$$

$$= \min \{ 244, 414, 158 \}$$

$$= 158$$

	1	2	3	4
1	0	1	1	3
2		0	2	3
3			0	3
4				0

cost table

m

	1	2	3	4
1	0	120	88	158
2		0	48	104
3			0	84
4				0

$$m[i,j] = \min_{i \le k \le j} \{ m[i,k] + m[k+1,j] + p_{i-1} p_k p_j \} \quad \text{if } i = j$$

Longest Common Subsequence

~~Applications:~~ Formulation.

$$LCS(m,n) = \begin{cases} 0 & \text{if } m=0 \text{ or } n=0 \\ 1 + LCS(m-1, n-1) & \text{if } x[m] == Y[n] \\ \max \begin{cases} LCS(m, n-1) \\ LCS(m-1, n) \end{cases} & \text{if } x[m] \neq Y[n] \end{cases}$$

X : A G G T A B   m = 6
Y : G X T X A Y B   n = 7

$1 + LCS(m-1, n-1)$

X : $\boxed{\text{A B C D H F}}$.   Case 1:
Y : $\boxed{\text{A C D F}}$ K

Case 1: $LCS(m, n-1)$
Case 2: $LCS(m-1, n)$

X : $\boxed{\text{A B C D H}}$ F   Case 2:
Y : $\boxed{\text{A C D F K}}$

Problem 1: X: A B C D   Y: A E D F

Step 1: Create a tabulation with $x(m)$ as rows; $Y(n)$ as col's. Place appropriate 0's

		0	1	2	3	4
		0	A	B	C	D
0	0	0	0	0	0	0
1	A	0				
2	E	0				
3	D	0				
4	F	0				

→ $x(m)$

$Y(n)$

Step 2:

LCS (0,0) = 0

LCS (1,1)

= 1 + LCS (0,0)

= 1 + 0

= 1

		0	1	2	3	4
		0	A	B	C	D
0	0	0	0	0	0	0
1	A	0				
2	E	0				
3	D	0				
4	F	0				

Enter '1' in A cell

Check for next
ie. LCS (1,2)
element A and B
are not matching

		0	1	2	3	4
		0	A	B	C	D
0	0	0	0	0	0	0
1	A	0	1	1	1	1
2	E	0	1	1	1	1
3	D	0	1	1	1	2
4	F	0	1	1	1	$\boxed{2}$

longest common subsequence

case 1

LCS (1,2)

= ~~1+~~ LCS (1,1)

= ~~1+~~ $\boxed{1}$.

max ?

case 2

LCS (1,2)

= LCS (0,2) = $\boxed{0}$

LCS (3,1) ⇒ LCS (3,0) ⇒ 0 } max ⇒ 1
⇒ LCS (2,1) ⇒ 1

LCS (2,1) ⇒ LCS (2,0) ⇒ 0 } max ⇒ 1
LCS (1,1) ⇒ 1

LCS (4,1) ⇒ LCS (4,0) ⇒ 0 } max ⇒ 1
LCS (3,1) ⇒ 1

$LCS(2,2) \Rightarrow LCS(2,1) \Rightarrow 1$
$LCS(1,2) \Rightarrow 1$ } max 1

$LCS(3,2) \Rightarrow LCS(3,1) \Rightarrow 1$
$LCS(2,2) \Rightarrow 1$ } max = 1

$LCS(4,2) \Rightarrow LCS(3,2) \Rightarrow 1$
$LCS(4,1) \Rightarrow 1$ } max = 1

$LCS(1,3) \Rightarrow LCS(1,2) \Rightarrow 1$
$LCS(0,3) \Rightarrow 0$ } max = 1

$LCS(2,3) \Rightarrow LCS(2,2) \Rightarrow 1$
$LCS(1,3) \Rightarrow 1$ } max = 1

$LCS(3,3) \Rightarrow LCS(2,3) \Rightarrow 1$
$LCS(3,2) \Rightarrow 1$ } max = 1

$LCS(4,3) \Rightarrow LCS(3,3) \Rightarrow$
$LCS(4,2) \Rightarrow$ } Not applicable as strings are matching

$LCS(4,3) \Rightarrow 1 + (3,2) \Rightarrow 1 + 1 \Rightarrow 2$

$LCS(1,4) \Rightarrow LCS(1,3) \Rightarrow 1$
$LCS(0,4) \Rightarrow 0$ } max = 1

$LCS(2,4) \Rightarrow LCS(1,4) \Rightarrow 1$
$LCS(2,3) \Rightarrow 1$ } max = 1

$LCS(3,4) \Rightarrow LCS(3,3) \Rightarrow 1$
$LCS(2,4) \Rightarrow 1$ } max = 1

$LCS(4,4) \Rightarrow LCS(4,3) \Rightarrow 2$
$LCS(3,4) \Rightarrow 1$ } max = 2

Note: Every element has a left and a diagonal
if two string are not same then
simply consider max of 'left' & 'above'
if ~~2~~ strings are same then 1 + (left & above max)

	0	1	2	3	4
	O	A	B	C	D
O	0	0	0	0	0
A	0	1	1	1 ?	1
E	0	1 ?	1	1	1
D	0	1	1	1	2
F	0	1	1	1	2

Go to D ⟺ F check same string or not → 2
Where does 2 comes from? (left and above)
left: 1 above: 2
Which is the max? '2' go ↑

Now check D ⟺ D, same string? yes '2'
Diagonally back ↖ $\boxed{D}$

Now check C ⟺ E, ~~str~~ same string? No '1'
Where does '1' comes from? left: 1, above: 1
go left and above both.

Now check C ⟺ A, same string? No '1'.
left: 1 and above '0' . consider max i.e left ←

Now check B ⟺ A, same string? No '1'
left: 1 above '0' . consider max i.e left ←

Now check A ⟺ A same string? Yes Diagonally back $\boxed{A}$

Check B ⟺ E, compute ?

$\boxed{AD}$ LCS ✓