

Definition

$\text{TIME}(t(n)) = \{L \mid L \text{ is a language decided by an } O(t(n)) \text{ time TM.}\}$

Class of P

$$P = \bigcup_{K} \text{TIME}(n^K)$$

P - P contains all the problems that can be solved / decided in Polynomial Time.

For example Sorting $\in \text{TIME}(n \log n)$
 PATH $\in \text{TIME}(n^2)$
 binary Search $\in \text{TIME}(\log n)$

Dynamic programming

- An efficient technique that can be used to solve problems that can be divided into smaller subproblems.

Matrix Multiplication

Given M_1, M_2, \dots, M_n , compute the product $M_1 M_2 M_3 \dots M_n$, where M_i has dimensions $d_{i-1} \times d_i$ (or M_i contains d_{i-1} rows and d_i columns)

Objective: minimize the total scalar multiplications.

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}_{2 \times 2} \begin{pmatrix} e \\ f \end{pmatrix}_{2 \times 1} = \begin{pmatrix} ae+df \\ ce+df \end{pmatrix}_{2 \times 1}$$

$$\text{Ex: } A_{n \times n} \cdot B_{n \times n} \cdot X_{n \times 1} =$$

$$(A \cdot B)_{n \times n} \cdot X_{n \times 1} \Rightarrow = n^3 + n^2 = O(n^3)$$

$$A_{n \times n} \cdot (B \cdot X)_{n \times 1} \Rightarrow n \times n \times 1 + n \times n \times 1 \\ = n^2 + n^2 \\ = O(n^2)$$

'Rephrase problem':

Parenthesize the product $M_1 M_2 \dots M_n$ in a way to minimize the # of scalar multiplication.

$$M_1 M_2 M_3 \dots \underbrace{\dots}_{\text{p}} \dots M_n$$

$$\text{Ex: } M_1 = 20 \times 10$$

$$M_2 = 10 \times 50$$

$$M_3 = 50 \times 5$$

$$M_4 = 5 \times 30$$

$$M_1 \times M_2 \times M_3 \times M_4$$

$$((M_1 M_2) (M_3 M_4)) = 47500$$

$$M_1 M_2 = 20 \times 10 \times 50 = 10000$$

$$M_3 M_4 = 50 \times 5 \times 30 = 7500$$

$$(M_1 M_2)_{20 \times 50} \cdot (M_3 M_4)_{50 \times 30} = 20 \times 50 \times 30 \\ = 30000 \\ (+) \\ = 47500$$

$$(M_1 (\underline{M_2 M_3}) M_4)$$

$$M_2 M_3 - 10 \times 50 \times 5 = 2500$$

$$M_1 \cdot ()_{10 \times 5} - 20 \times 10 \times 5 = 1000$$

$$()_{20 \times 5} \cdot M_4_{5 \times 30} - 20 \times 5 \times 30 = 3000$$

+

6880

$$(M_1 M_2 M_3 \dots M_k) \uparrow (M_{k+1} \dots M_n)$$

$P(n)$ = # of alternative parenthesis
sations of n matrices

$$P(k) \quad P(n-k)$$

$$P(n) = \begin{cases} 1 & \text{if } n=1 \\ \sum_{k=1}^{n-1} P(k) P(n-k), & \text{if } n \geq 2 \end{cases}$$

$P(n)$ - n^{th} catalan number

$$P(n) = \frac{1}{n} \binom{2(n-1)}{n-1} \geq \frac{4^{n-1}}{2n-1} = \Omega\left(\frac{4^n}{n^2}\right)$$

$$n=10 ; P(n) \geq \Omega\left(\frac{4^{10}}{10^2}\right) \approx 10^4$$

$$n=20 ; P(n) \geq \Omega\left(\frac{4^{20}}{20^2}\right) \approx 10^9$$

Dynamic programming solution

$$(M_1 M_2 M_3 \dots M_K) (M_{K+1} \dots M_n)$$

- $m[i, j]$ is the number of multiplications performed using optimal parenthesization of

$$M_i M_{i+1} \dots M_j$$

$$M_i M_{i+1} \dots M_k M_{k+1} \dots M_j$$

$\downarrow d_{i-1} \times d_i$ $\downarrow d_{k-1} \times d_k$ $\downarrow d_k \times d_{k+1}$ $\downarrow d_{j-1} \times d_j$

$$[m[i, j] = \min_k \{ m[i, k] + m[k+1, j] + d_{i-1} \times d_k \times d_j\}$$

$$m[i, i] = 0$$

$m[1, n]$ contains the optimal number of scalar multiplications for the product.

$$M_1 = 20 \times 10$$

$$M_2 = 10 \times 50$$

$$M_3 = 50 \times 5$$

$$M_4 = 5 \times 30$$

	1	2	3	4
1	0	<u>10000</u>	3500	0
2	0	2500	0	0
3	0	0	7500	0
4	0	0	0	0

$$m[1,2] = \min_k [m[1,k] + m[k+1,2]] +$$

$$\sum_{k=1}^K m[1,1] + m[2,2]$$

$$+ d_{i-1} \times d_k \times d_j$$

$$20 \times 10 \times 50$$

$$m[2,3] = m[2,2] + m[3,3] +$$

$$d_{i-1} \times d_k \times d_j$$

$$10 \times 50 \times 5$$

$$\approx 2500$$

$$m[3,4] = m[3,3] + m[4,4] + \\ d_{i,1} \times d_k \times d_j \\ 50 \times 5 \times 30$$

$$m[1,3] = \min \left\{ \begin{array}{l} \underset{k=1}{m[1,1] + m[2,3]} \\ \quad \quad \quad + d_{1,1} \times d_k \times d_3 \\ 0 + 2500 + 20 \times 10 \times 5 \\ \quad \quad \quad = 3500 \end{array} \right.$$

$$\left. \begin{array}{l} k=2 \\ m[1,2] + m[3,3] + \\ \quad \quad \quad d_{1,1} \times d_k \times d_3 \\ 10000 + 0 + \\ \quad \quad \quad 20 \times 50 \times 5 \\ \quad \quad \quad = 15000 \end{array} \right\}$$

i	1	2	3	4	
j	0	<u>10000</u>	3500	6500	$M_1 = 20 \times 10$
2	0	2500	4000	$M_2 = 10 \times 50$	
3	0	7500		$M_3 = 50 \times 5$	
4	0			$M_4 = 5 \times 30$	

$$m[2,4] = \min \left\{ \begin{array}{l} K=2 \\ m[2,2] + m[3,4] \\ + d_{i-1} \times d_K \times d_j \\ 0 + 7500 \\ + 10 \times 50 \times 30 \\ 7500 + 15000 \\ \\ K=3 \\ m[2,3] + m[4,4] \\ + d_{i-1} \times d_K \times d_j \\ 2500 + 0 \\ + 10 \times 5 \times 30 \\ 2500 + 1500 \\ 4000 \end{array} \right\}$$

$$\begin{aligned} &K=2 \\ &m[2,2] + m[3,4] \\ &+ d_{i-1} \times d_K \times d_j \\ &0 + 7500 \\ &+ 10 \times 50 \times 30 \\ &\textcircled{7500 + 15000} \\ \\ &K=3 \\ &m[2,3] + m[4,4] \\ &+ d_{i-1} \times d_K \times d_j \\ &2500 + 0 \\ &+ 10 \times 5 \times 30 \\ &2500 + 1500 \\ &\textcircled{4000} \end{aligned}$$