Lecture 02 Date: 4 Jan 24

▷ Create an array Fib[0 ... n]

$Fib[0] \leftarrow 0$

$Fib[1] \leftarrow 1$

<u>for</u> $i \leftarrow 1$ <u>to</u> n

<u>do</u> $Fib[i] \leftarrow Fib[i-1] + Fib[i-2]$

▷ Create $T[1 \dots n][0 \dots t]$

<u>for</u> $i \leftarrow 1$ <u>to</u> n
    <u>do</u> $T[i][0] \leftarrow$ True

<u>for</u> $j \leftarrow 1$ <u>to</u> t
    <u>do</u> <u>if</u> $A[1] = j$
        <u>then</u> $T[1][j] \leftarrow$ True
        <u>else</u> $T[1][j] \leftarrow$ False.

for $i \leftarrow 2$ **to** n

  do for $j \leftarrow 1$ **to** t

    do **if** $A[i] \leq j$

      then $T[i][j] \leftarrow T[i-1][j-A[i]] \vee T[i-1][j]$

      else: $T[i][j] \leftarrow T[i-1][j]$

Matrix Chain Multiplication

Example:-

$$\big(A_1 \overbrace{\underbrace{(A_2\, A_3)}_{C}}^{100 \times 50}\big)$$

$10 \times 100 \qquad 100 \times 5 \qquad 5 \times 50$

$100 \times 5 \times 50 = 25000$

$A_1 C \rightarrow 10 \times 100 \times 50 = 50000$

Total 75000

$C = A\,B$

$A: s \times t \qquad B: t \times r$

$$c_{ij} = \sum_{k=1}^{t} a_{ik} b_{kj}$$

Total # mult^n $= s\,t\,r$.

$((A_1 A_2) A_3)$ 7500

Input:- A sequence of matrices $A_1, \ldots, A_n$

Find:- The "best" way to multiply these matrices
in order

The one that minimizes # mult

$$\prod_{i=1}^{n} A_i$$

A_i has dimension $p_{i-1} \times p_i$

p $\quad p_0, \dots, p_n.$

A_1

$P(n) = \#$ ways to multiply n matrices

$P(1) = 1 \qquad P(3) = 2$

$P(2) = 1 \qquad P(4) = 5$ (?)

$((A_1 A_2)(A_3 A_4))$

$((A_1 (A_2 A_3)) A_4)$

$(A_1 ((A_2 A_3) A_4))$

$(A_1 (A_2 (A_3 A_4)))$

$(((A_1 A_2) A_3) A_4)$

$$P(n) = \sum_{k=1}^{n-1} P(k) P(n-k)$$

$P(n)$ Catalan number

$\downarrow$

$$\frac{1}{n}\binom{2n}{n}$$

$$P(n) = \Omega\left(\frac{4^n}{n^{3/2}}\right)$$

$$P(n) > 2^n$$

$m[i, j]$ = optimal cost of multiplying $A_i, \ldots, A_j$

$m[i, i] = 0$ A_i

$p_{i-1} \times p_k$ $\quad$ $p_k \times p_j$

$$\overbrace{\underbrace{A_i \cdots A_k}_{p_{i-1} \times p_i}}\ \Big|\ \overbrace{\underbrace{A_{k+1}\ A_j}}$$

$$m[i,k] + m[k+1,j] + p_{i-1}\, p_k\, p_j$$

$$\underbrace{m[i,j]} = \min_{i \le k < j} \left\{ \underbrace{m[i,k]} + \underbrace{m[k+1,j]} + p_{i-1}\, p_k\, p_j \right\}$$

for $i \leftarrow 1$ **to** n
  do $m[i,i] \leftarrow 0$ $O(n)$

for $l \leftarrow 2$ **to** n [size of the window]
  do for $i \leftarrow 1$ **to** $n-l+1$
    do $j \leftarrow i+l-1$
      $m[i,j] \leftarrow \infty$
      for $k \leftarrow i$ **to** $j-1$
        do $q \leftarrow m[i,k] + m[k+1,j] + p_{i-1} p_k p_j$
          if $q < m[i,j]$
            then $m[i,j] \leftarrow q$
              $s[i,j] \leftarrow k$

Runtime

$O(n^3)$

HW:- ① Find the optimal multiplication strategy
② Implement.

Optimal Substructure property:-

$A_1 \dots A_n$

$A_1 \dots A_k \mid A_{k+1} \dots A_n$

$((A_1 (A_2 A_3)) \mid (A_4 (A_5 A_6)))$

An optimal solⁿ to a problem contain within it the optimal solⁿ to subproblems.

Overlapping subproblem property.

$$[A_1 (A_2 A_3) A_4] A_5 .$$