Matrix Exponentiation :-

Prerequisites :-

(1) Matrix multiplication :-

Let $A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1k} \\ \vdots & & & \vdots \\ a_{m1} & \cdots & \cdots & a_{mk} \end{bmatrix}_{m \times k}$

$B = \begin{bmatrix} b_{11} & b_{12} & & b_{1n} \\ \vdots & & & \vdots \\ b_{k1} & \cdots & \cdots & b_{kn} \end{bmatrix}_{k \times n}$

Let $C = A \times B$

Order of $C = m \times n$

$$C_{ij} = \sum_{r=1}^{k} a_{ir} \times b_{rj}$$

eg: $m, n, k = 3.$

$$\begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & & & \\ 2 & & & (2,3) \\ 3 & & & \end{array}$$

$C_{23} = a_{21} \times b_{13} + a_{22} \times b_{23} + a_{23} \times b_{33}$

$\rightarrow \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix}$

Time Complexity :-

Total elements $\times$ Time taken to calculate each element.

$(m \times n) \times k$

If $m = n = k = K$ (i.e. square matrix)

$$\boxed{\text{Time Complexity} = O(K^3)}$$

(2) Calculating A^n, order $= K \times K$

$\uparrow$

matrix

$$A^n \rightarrow \begin{cases} A^{n/2} \times A^{n/2}, & n = \text{even} \\ A \times A^{n/2} \times A^{n/2}, & n = \text{odd} \end{cases}$$

Time complexity :-

$val = pow(A, n/2)$

$A^n = val \times val$ $n \rightarrow$ even

$A^n = A \times val \times val$ $n \rightarrow$ odd.

Scanned with CamScanner

$$T(n) = T(n/2) + O(k^3)$$

finding A^n | finding $A^{n/2}$ | multiplying $A^{n/2} * A^{n/2}$

$\log_2 n$ steps

$$T(n/2) = T(n/4) + O(k^3)$$

$$T(1)$$

$$\boxed{\text{Time complexity} = O(k^3 \log_2 n)}$$

Main Topic : Matrix Exponentiation

The concept of matrix exponentiation in it's most general form is very useful in solving questions that involve calculating nth term of a linear recurrence relation in time of the order of log n

E.g: $f(n) = f(n-1) + f(n-2)$ ← Linear reccurence relation of fibonacci.

We can calculate nth fibonacci in log n time

$$K^{th} \text{ State} \xrightarrow{\text{Matrix } M} (K+1)^{th} \text{ State}$$

$$M \times \begin{bmatrix} f(n) \\ f(n-1) \\ \vdots \\ f(n-K) \end{bmatrix} = \begin{bmatrix} f(n+1) \\ f(n) \\ \vdots \\ f(n-K+1) \end{bmatrix}$$

eg: fibonacci :-

Linear Reccurence Relation : $f(n) = f(n-1) + f(n-2)$ — ①

$$\underset{\substack{\uparrow \\ \text{order } 2\times 2}}{M} \times \begin{bmatrix} f(n) \\ f(n-1) \end{bmatrix}_{2\times 1} = \begin{bmatrix} f(n+1) \\ f(n) \end{bmatrix}_{2\times 1}$$

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} f(n) \\ f(n-1) \end{bmatrix} = \begin{bmatrix} f(n+1) \\ f(n) \end{bmatrix}$$

$$a \times f(n) + b\, f(n-1) = f(n+1)$$

from ①, $a=1$ $b=1$

$$c \times f(n) + d\, f(n-1) = f(n)$$

By obsvⁿ, $c=1$ $d=0$

$\therefore \quad M = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$

Now,

$$M \times \begin{bmatrix} f(n) \\ f(n-1) \end{bmatrix} = \begin{bmatrix} f(n+1) \\ f(n) \end{bmatrix} \quad - (1)$$

$$M \times \begin{bmatrix} f(n+1) \\ f(n) \end{bmatrix} = \begin{bmatrix} f(n+2) \\ f(n+1) \end{bmatrix}$$

from (1),

$$M \times M \times \begin{bmatrix} f(n) \\ f(n-1) \end{bmatrix} = \begin{bmatrix} f(n+2) \\ f(n+1) \end{bmatrix}$$

$$M^2 \times \begin{bmatrix} f(n) \\ f(n-1) \end{bmatrix} = \begin{bmatrix} f(n+2) \\ f(n+1) \end{bmatrix}$$

$$M^K \times \begin{bmatrix} f(n) \\ f(n-1) \end{bmatrix} = \begin{bmatrix} f(n+K) \\ f(n+K-1) \end{bmatrix}$$

For N^{th} fibonacci, put $n = 1$, $K = N-1$

$$M^{N-1} \times \begin{bmatrix} f(1) \\ f(0) \end{bmatrix} = \begin{bmatrix} f(1+N-1) \\ f(N-1) \end{bmatrix}$$

$$M^{N-1} \times \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} f(N) \\ f(N-1) \end{bmatrix}$$

Say $M^{N-1} = \begin{bmatrix} x & y \\ a & b \end{bmatrix}$ $\left\{ \begin{bmatrix} x & y \\ a & b \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right\}$

Then

$$f(N) = x \times 1 + y \times 0$$

$$\boxed{f(N) = x}$$

$\therefore$ $f(N)$ will be $\underline{\underline{(0,0)}}^{th}$ element (0-based indexing) the M matrix.

Scanned with CamScanner