

Maths Concepts & Qns



∞  → codestorywithMIK

X  → CSwithMIK

 → codestorywithMIK

Motivation :-

Do you want to grow ???



- ↳ Come out of your comfort zone.
- ↳ Stop running from tough/new topics

MIX

Remember, everything seems difficult in the beginning,
slowly you start realising how easy it was...

Matrix Exponentiation

Powerful optimization technique, especially
useful in problems involving linear recurrences.

For Example:-

$$F(n) = F(n-1) + F(n-2)$$

DP Concepts & Qns : Convert Story To Code

by codestorywithMIK

Playlist • 34 videos • 2,30,824 views

The main motive of this playlist is to understand each techniq ...more

VIDEO-1

Fibonacci Number | Recur + Memo | Bottom Up | DP Concepts & Qns - 2 | Leetcode-509

codestorywithMIK

$$F(n) = F(n-1) + F(n-2)$$

✓✓

Recursive soln $\rightarrow T.C = O(2^n)$

Bottom up $\rightarrow T.C = O(n)$

Linear Recurrence Relation

So, we can further improve our

solution from $O(n)$ to $\log(n)$

How ???

O O
 I
 O

$$F(n) = F(n-1) + F(n-2)$$

$(n-2)^{th}$ $(n-1)^{th}$ $(n)^{th}$ $(n+1)^{th}$ $(n+2)^{th} \dots$

$$\begin{bmatrix} F(n) \\ F(n-1) \end{bmatrix}_{2 \times 1} = T * \begin{bmatrix} F(n-1) \\ F(n-2) \end{bmatrix}_{2 \times 1}$$

$$\begin{bmatrix} F(n) \end{bmatrix} = \begin{bmatrix} a * F(n-1) + b * F(n-2) \end{bmatrix}$$

The diagram illustrates the derivation of a recurrence relation for the Fibonacci sequence. At the top, two green brackets labeled 2×1 enclose terms: $F(n-1)$ on the left and $C * F(n-1) + d * F(n-2)$ on the right. Below, a large bracket labeled 2×1 encloses the equation $F(n) = a * F(n-1) + b * F(n-2)$. The term $F(n)$ is crossed out with a large black circle. Instead, the equation is shown as $F(n) = a * F(n-1) + b * F(n-2)$. Arrows point from the terms $F(n-1)$ and $F(n-2)$ in this equation to the corresponding terms in the top brackets. To the right, the equation is labeled $\rightarrow ①$. Below, another equation is shown: $1 * F(n-1) = C * F(n-1) + d * F(n-2)$, with arrows pointing from the terms $F(n-1)$ and $F(n-2)$ to the corresponding terms in the top brackets. This equation is labeled $\rightarrow ②$.

$$a = 1 \quad , \quad b = 1$$

$$c = 1 \quad , \quad d = 0$$

$$T = \begin{bmatrix} 1 & 1 \\ -1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} F(n) \\ F(n-1) \end{bmatrix}_{2 \times 1} = T * \begin{bmatrix} F(n-1) \\ F(n-2) \end{bmatrix}_{2 \times 1}$$

$$F(0) = 0$$

$$F(1) = 1$$

$$n=2$$

$$F(n) = F(2) ??$$



$$\begin{bmatrix} F(2) \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} * \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} (1*1) + (0*0) \\ (1*1) + (0*0) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} F(2) \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \text{Result?}$$

Result? $[0][0] \rightarrow 1$

$$\begin{bmatrix} F(2) \\ F(1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}}_T * \begin{bmatrix} F(1) \\ F(0) \end{bmatrix}$$

$$T = \begin{bmatrix} 1 & 1 \end{bmatrix}$$

$$T^2 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$$

$n=3$

$$T^2 * \begin{bmatrix} F(1) \\ F(0) \end{bmatrix}$$

$$= \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} * \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} F(3) \\ F(2) \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \quad \text{3rd Fibonacci no.}$$

$n=3$

$$T^2 * \begin{bmatrix} F(1) \\ F(0) \end{bmatrix}$$

$T^{(0)}$

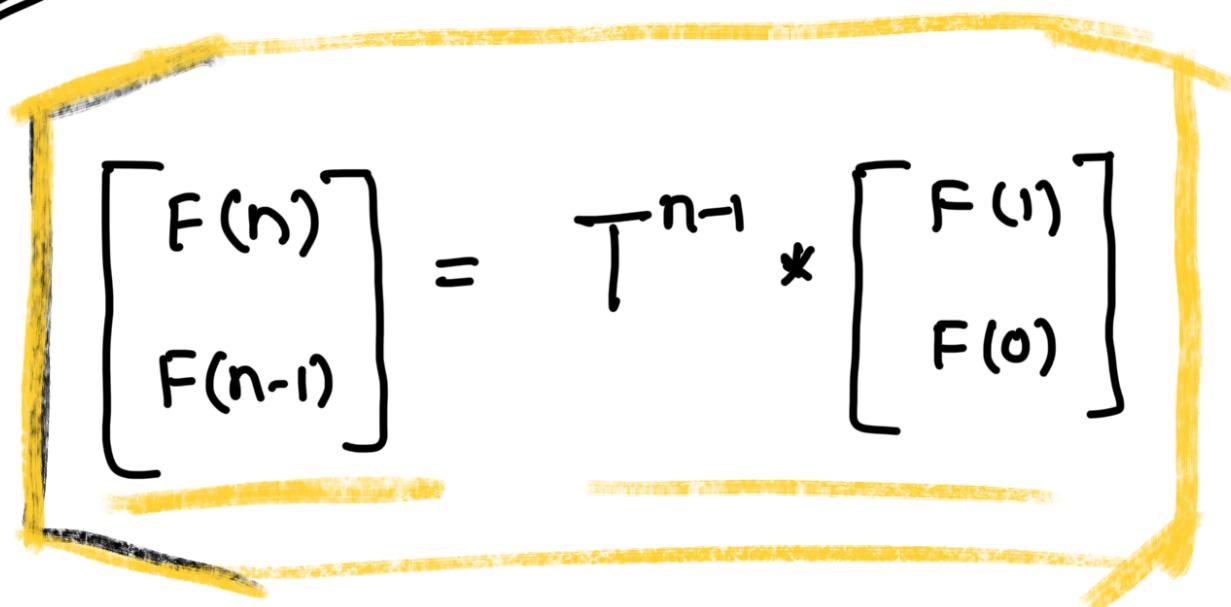
$n=4$

$$T^3 * \begin{bmatrix} F(1) \\ F(0) \end{bmatrix}$$

$n=5$

$$T^4 * \begin{bmatrix} F(1) \\ F(0) \end{bmatrix}$$

Generalise:-



$$\begin{bmatrix} F(n) \\ F(n-1) \end{bmatrix} = T^{n-1} * \begin{bmatrix} F(1) \\ F(0) \end{bmatrix}$$

Degree Of a

Recurrence

It tells you how many previous terms it depends on.

Example :- $F(n) = \underbrace{F(n-1) + F(n-2)}$
Two terms

So, it's a "Degree 2" relation.

$$\begin{bmatrix} F(n) \\ F(n-1) \end{bmatrix} = T * \begin{bmatrix} F(n-1) \\ F(n-2) \end{bmatrix}$$

Example :-

$$F(n) = F(n-1) + F(n-2) + F(n-3)$$

degree = 3

$$\begin{bmatrix} F(n) \\ F(n-1) \\ F(n-2) \end{bmatrix} = T \begin{bmatrix} F(n-1) \\ F(n-2) \\ F(n-3) \end{bmatrix}$$

T

3×3

$$T = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Where is
exponentiation ???

1
1

$F(n) \rightarrow \log(n) ??????$

Remember, we had to find T^{n-1}
(Binary Exponent)

$$n = 6$$

$$\cdot T^5$$

$$\begin{bmatrix} & \\ & \end{bmatrix}_{2 \times 2}$$

$$\log(n-1) \approx \log(n)$$

```
int findPower (base, exponent) {
```

```
    if (exponent == 0) {
```

```
        return 1;
```

```
}
```

```
    half = findPower (base, exponent/2);
```

```
    result = half * half;
```

```
    if (exponent % 2 == 1) {
```

```
        result = (result * base);
```

```
}
```

```
return result;
```

```
Matrix matrixExp (Matrix base , int exponent){
```

```
    if (exponent == 0) {
```

```
        make an identity matrix I
```

```
        return I;
```

```
    half = matrixExp (base, expon/2);
```

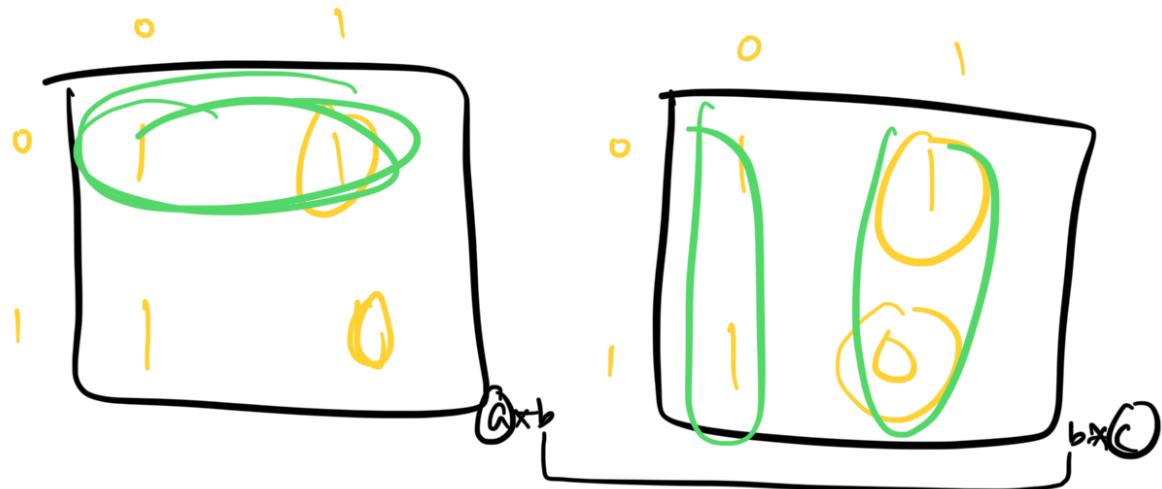
```
    result = matrixMulti(half, half);
```

```
    if (exponent/2 == 1) {
```

```
        result = matrixMulti(result, base);
```

```
}
```

return result;



Matrix matrixMult (Matrix A_{2x2}, Matrix B_{2x2}) {
 }

Matrix C_{2x2} ;

for (i=0; i<2; i++) {

 for (j=0; j<2; j++) {

 for (k=0; k<2; k++) {

$$C[i][j] = (C[i][j] + A[i][k] * B[k][j]) ;$$

}

}

}

$$n = 6$$

$F(n)$

$F(6)$

$$T \cdot C = \log(n)$$

$\text{solve}(n)$

$F(n)$

$$\begin{bmatrix} F(n) \\ F(n-1) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}_{2 \times 2}^{n-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\text{matrixExp}(T, n-1) * \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$



return result [0][0] ✓

- { * Matrix Exponentiation.
- * Matrix Multiplication.

$$\frac{P \cdot O \cdot T \cdot D}{\text{let } \approx 3337}$$

(.) Recc. relation

(.) degree $\rightarrow T_{xxx}$

