Catalan Number

$C_0 = 1$

$C_1 = 1$

$C_2 = C_0 \cdot C_1 + C_1 \cdot C_0 = 1.1 + 1.1 = 2$

$C_3 = C_0 \cdot C_2 + C_1 \cdot C_1 + C_2 \cdot C_0 = 1.2 + 1.1 + 2.1$
$= 5.$

$C_4 = C_0 C_3 + C_1 \cdot C_2 + C_2 C_1 + C_3 \cdot C_0 = 1.5 + 1.2 + 2.1 + 5.1$
$= 14$

$C_5 = C_0 \cdot C_4 + C_1 \cdot C_3 + C_2 \cdot C_2 + C_3 \cdot C_1 + C_4 \cdot C_0$
$= 1.14 + 1.5 + 2.2 + 5.1 + 14.1 = 42$

$C_N = ?$

$C_0 \cdot C_{N-1} + C_1 \cdot C_{N-2} + C_2 \cdot C_{N-3} + C_3 \cdot C_{N-4} + C_4 \cdot \_ + C_5 \cdot \_ \; - - \_\_\_ \; C_{N-1} \cdot C_0$

C_i	* C_{N-i-1}
0	N-1
1	N-2
2	N-3
3	N-4
4	N-5
5	N-6
|	|
|	|
N-1	C_0

$$\boxed{C_n = \sum_{i=0}^{N-1} C_i * C_{N-i-1}}$$

pseudo-code.

int dp[N+1] //

meaning of dp[i] → value of C_i at i^{th} index.

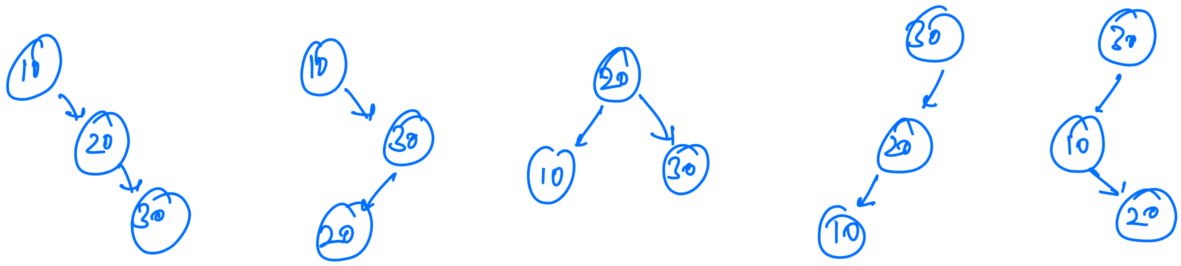
```
dp[0] = 1
dp[1] = 1
for( i=2 ; i <= N ; i++){
        val = 0 ;
        for( j=0 ; j < i ; j++){
                val += dp[j] * dp[i-j-1] ;
        }
        dp[i] = val
}
return dp[N] ;
```

$T.C \to O(N^2)$

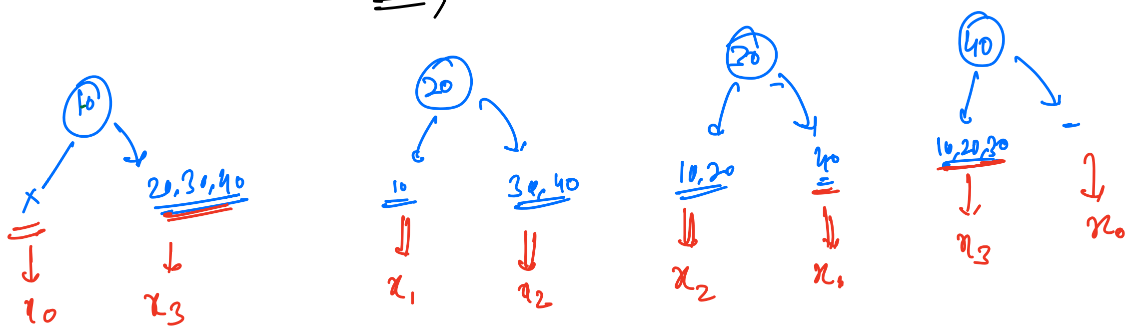
$S.C \to O(N)$

Q: Given N, count total no. of unique BST's you can form with N distinct nodes.

N=3. 10, 20, 30



N=4, 10, 20, 30, 40



$$\left\{ x_4 = x_0 \cdot x_3 + x_1 \cdot x_2 + x_2 \cdot x_1 + x_3 \cdot x_0 \right\}$$

$$\left\{ x_5 = x_0 \cdot x_4 + x_1 \cdot x_3 + x_2 \cdot x_2 + x_3 \cdot x_1 + x_4 \cdot x_0 \right\}$$

$\therefore$ { No. of unique B.S.T formed with N distinct nodes = N^{th} Catalan Number }

No. of unique configuration created by N parenthesis pairs?

N=1 ( ) 1

N=2 ( ) ( ) , (( )) 2

N=3. ( ) ( ) ( ) , ( ) (( )) , (( )) ( ) , ((( ))) , (( ) ( )) 5

N=4

0 (3) $\Rightarrow$ $x_0 \cdot x_3$ +

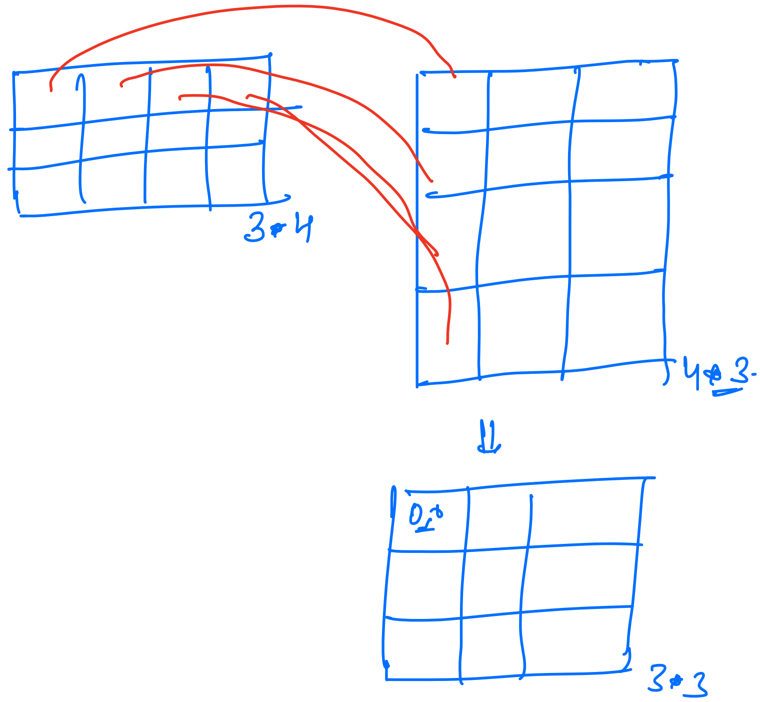
1 (2) $\Rightarrow$ $x_1 \cdot x_2$ +

2 (1) $\Rightarrow$ $x_2 \cdot x_1$ +

3 (0) $\Rightarrow$ $x_3 \cdot x_0$

x_4 = 4th Catalan no.

Matrix Chain Multiplication



total no. of multiplication = $3*3*4 = 36$.

$$[\quad]_{a*b} \quad [\quad]_{b*c}$$

total no. of multiplications = $a*b*c$.

M_1 M_2 M_3

$(3*5)$ $(5*4)$ $(4*7)$

$60 + 84 = \underline{144}$

1. $\left([\ ]_{3*5} \cdot [\ ]_{5*4} \right) \cdot [\ ]_{4*7} \Rightarrow [\ ]_{3*4} [\ ]_{4*7}$

2. $[\ ]_{3*5} \cdot \left([\ ]_{5*4} \cdot [\ ]_{4*7} \right) \Rightarrow [\ ]_{3*5} [\ ]_{5*7}$

$140 + 105 = \underline{245}$

Q) Given some matrices. Find min cost to multiply matrices.

Array of size N , N = 4

$\{ \ 3 \quad 5 \quad 7 \quad 4 \ \}$

$3*5$ $5*7$ $7*4$

matrix 1 → arr[0] * arr[1]

matrix 2 ⇒ arr[1] * arr[2]

matrix 3 ⇒ arr[2] * arr[3]

matrix i ⇒ arr[i-1] * arr[i]

matrix N-1 ⇒ arr[N-2] * arr[N-1]

$M_1 \quad M_2 \quad M_3 \quad M_4 \quad M_5$

$M_1 \quad M_2 \quad M_3 \quad M_4 \quad M_5$

$M_1 \quad M_2 \quad M_3 \quad M_4 \quad M_5$

$M_1 \quad M_2 \quad M_3 \quad M_4 \quad M_5$

8 - Matrices.

$M_1 \quad M_2 \quad M_3 \quad M_4 \quad M_5 \quad M_6 \quad M_7 \quad M_8$

Min

$[1-1] + [2-8]$ + $a[0] * a[1] * a[8]$

$C_1 + C_2$

$a[0]*a[1] \quad a[1]*a[8]$

$[1-2] + [3-8]$ + $a[0] * a[2] * a[8]$

$C_1 + C_2$

$a[0]*a[2] \quad a[2]*a[8]$

$[1-3] + [4-8]$ + $a[0] * a[3] * a[8]$

$[1-4] + [5-8]$ + $a[0] * a[4] * a[8]$

$[1-5] + [6-8]$ + $a[0] * a[5] * a[8]$

$[1-6] + [7-8]$ + $a[0] * a[6] * a[8]$

$[1-7] + [8-8]$ + $a[0] * a[7] * a[8]$

$[\ ]_{3*7} ([\ ]_{7*4} [\ ]_{4*11} [\ ]_{11*14} [\ ]_{14*6} [\ ]_{6*13})$

dimension of resultant matrix $\Rightarrow 7 * 13$.

min-cost to multiply matrices from i to j.

$$
\min \left\{
\begin{array}{lll}
\overset{K}{(i,i)} + (i+1,j) & + & \text{extra cost} \\
(i,i+1) + (i+2,j) & + & \text{"} \\
(i,i+2) + (i+3,j) & + & \text{"} \\
(i,i+3) + (i+4,j) & + & \text{"} \\
(i,i+4) + (i+5,j) & + & \text{"} \\
\vdots & & \\
(i,j-1) + (j,j) & + & \text{"}
\end{array}
\right.
$$

$$
\boxed{\text{min-Cost}(i,j) = \underset{k=i \text{ to } j-1}{\min} \left\{ \forall k \quad \text{min-Cost}(i,k) + \text{minCost}(k+1,j) + a[i-1] * a[k] * a[j] \right.}
$$

	m_1	m_2	m_3	m_4	m_5	m_6	m_7
10	3	9	6	11	2	5	7
0	1	2	3	4	5	6	7

j (above m_7)

$\text{minCost}(0,0) + \text{minCost}(1,7) + a[0] * a[1] * a[7]$

$\text{minCost}(0,1) + \text{minCost}(2,7) + a[0] * a[2] * a[7]$

$\vdots$

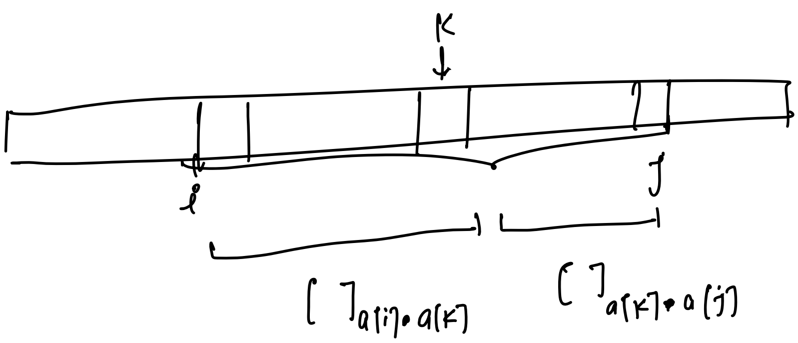
$\text{min Cost}(0,6) + \text{minCost}(7,7) + a[0] * a[6] * a[7]$

int dp[N][N];

//meaning dp[i][j] → min cost to multiply all the matrices from i to j.

```
int minCost( arr[], i, j){
    if ( i == j) { return 0}
    if (dp[i][j] != -1) { return dp[i][j] }
    cost = Integer.MAX ;
    for( k = i ; k < j ; k++){
        cost = Min( cost, minCost( i, k) +
                          minCost ( k+1, j) +
                          a[i] * a[k] * a[j] )
    }
    dp[i][j] = cost
    return cost
}
```

T.C → $O(N^3)$
S.C → $O(N^2)$



cost to multiply resultant matrices = a[i] * a[k] * a[j]

[# iterative.] # todo

Buy and sell Stocks - 1.

Given an arr which represents price of a stock on first N days.

At most 1 single transaction is allowed to you.

Buy → sell

Max profit?

max=7	max=6	max=6	max=6	max=6	max=4
7	1	5	3	6	4

arr[] = 7, 1, 5, 3, 6, 4

profit = ~~0~~ ~~3~~ 5

for an element, we are just looking for max element on its r.h.s.

idea → we need to find max element on r.h.s for every element.

pseudo-code:

```
profit = 0, max = arr[N-1]
for( i = N-2 ; i >= 0 ; i-- ){
        max = Max( max, arr[i])
        if ( max - arr[i] > profit){
              profit = max - arr[i]
        }
}
return profit;
```

$\boxed{\begin{array}{l} T.C \to O(N) \\ S.C \to O(1) \end{array}}$

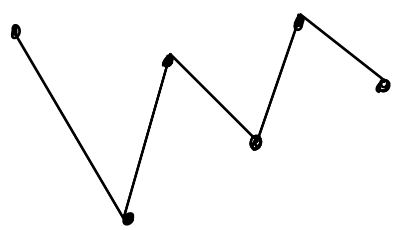
Buy & Sell Stocks – II

no limit on transactions.

→ you can't buy again until you sell the previous stock.

arr[] → [7 1 5 3 6 4]   profit = 7.

(1, 5) → 4   (3, 6) → 3



[Buy at dip & sell at greater value.]

arr[] → [2 4 6 8 12 10 15 20]

2 2 2 4   5 5

ans = 20

```
profit = 0
for ( i = 1 ; i < N ; i++) {
    if ( arr[i-1] < arr[i]) {
        profit += arr[i] - arr[i-1]
    }
}
```

[T.C → O(N)
S.C → O(1)]

(3) At most 2 transactions are allowed? → think

(4) At most K transactions are allowed?

10-15 problem → [memoization
tabulation]