Catalom Number:-

Complexity → $O(n^2)$

$C_0 \to 1$

$C_1 \to 1$

$C_2 \to 2$

$C_3 \to C_0 \cdot C_2 + C_1 \cdot C_1 + C_2 \cdot C_0 = 1*2 + 1*1 + 2*1 = 2+1+2 = 5$

$C_4 \to C_0 \cdot C_3 + C_1 \cdot C_2 + C_2 \cdot C_1 + C_3 \cdot C_0 = 1*5 + 1*2 + 2*1 + 5*1 = 5+2+2+5 = 14$

$C_5 \to C_0 \cdot C_4 + C_1 \cdot C_3 + C_2 \cdot C_2 + C_3 \cdot C_1 + C_4 \cdot C_0 = 1*14 + 1*5 + 2*2 + 5*1 + 14*1$

$= \underline{14} + \underline{5} + \underline{4} + \underline{5} + \underline{14}$

$= \underline{\underline{42}}$

$\boxed{C_n} \to C_0 \cdot C_{n-1} + C_1 \cdot C_{n-2} + C_2 \cdot C_{n-3} + \cdots \cdots + C_{n-1} \cdot C_0$

i ← n time ← for every i, internal or iternal

0	1	2	3	4	5
1	1	2	5	14	42

$\underline{14} + \underline{5} + \underline{4} + \underline{5} + \underline{14}$

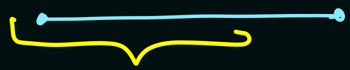
42

$k_1 = 0$

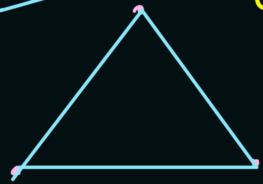
$k_2 = i-1$

Number of ways of triangulation :

$n = 2 = 1 \rightarrow C_0$

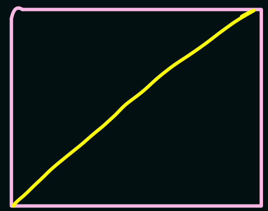


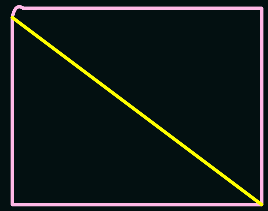
$n = 3 = 1 \rightarrow C_1$



n sides $\rightarrow$ $(n-2)$ catalan

$n = 4 = 2 \rightarrow C_2$

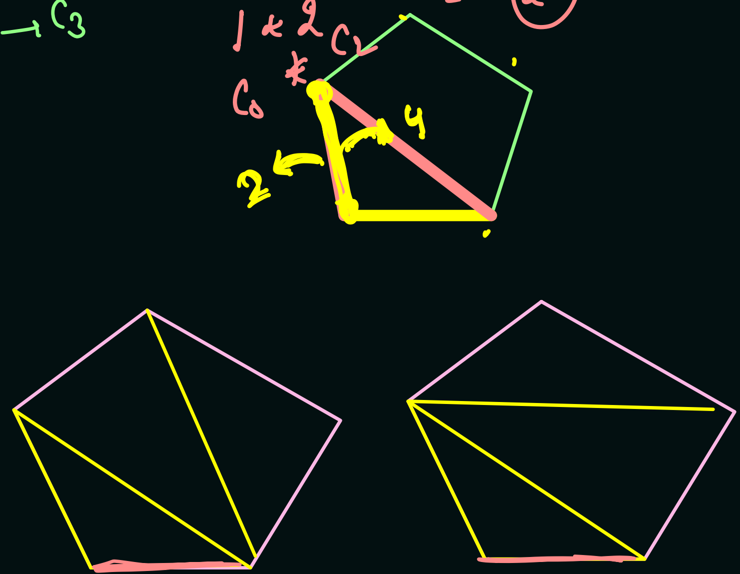




$n = 5 \rightarrow C_3$

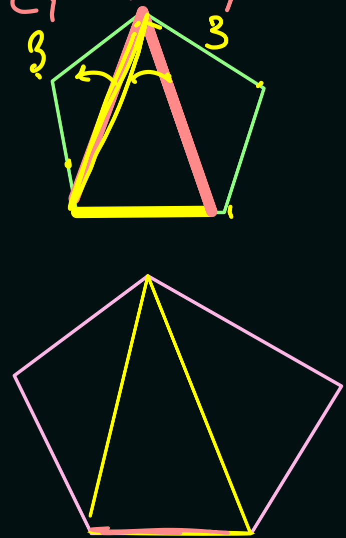
$1 * 2 = 2$

$C_0 * C_2$



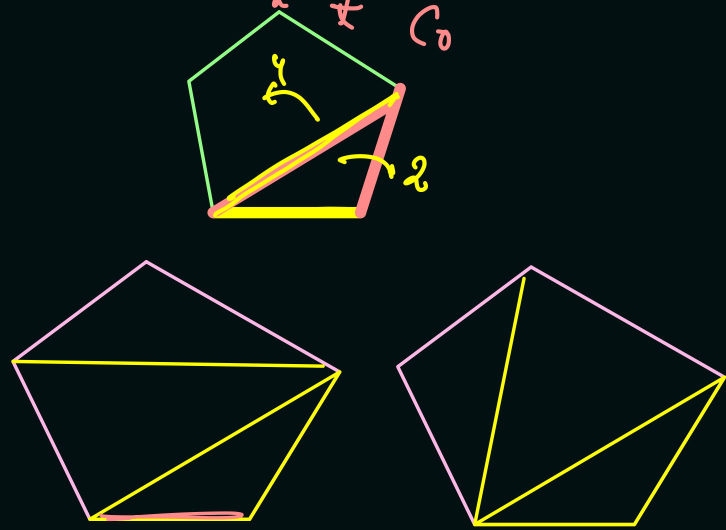
$1 * 1 = 1$

$C_1 * C_1$

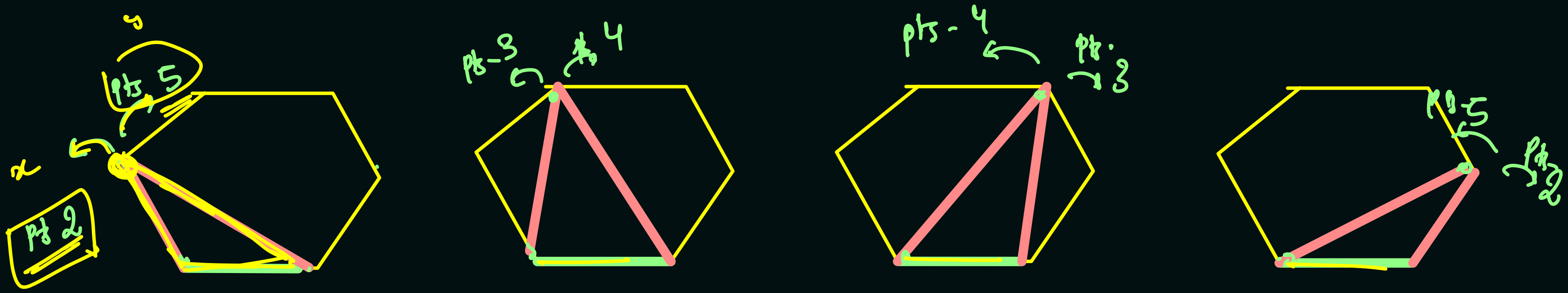


$2 * 1 = 2$

$C_2 * C_0$



$n = 6$ $C_4 \to 14$



$$C_4 = C_0 * C_3 + C_1 * C_2 + C_2 * C_1 + C_3 * C_0$$

$$1 * 5 + 1 * 2 + 2 * 1 + 5 * 1$$

Catalan Num

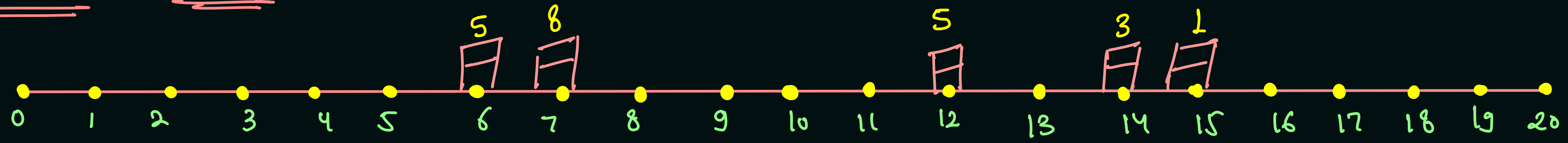
$$5 + 2 + 2 + 5 = 14$$

Highway Billboard :→

n → no. of billboard
m → Total distance of Highway.

min. distance = 5.

Method 1 (O(n²))



LIS →

6-5 7-8 12-5 14-3 15-1

} Max sum of subseq.

Condition → 't' unit separation distance.

t=5

5	8	10	11	9

i → * Max revenue generation if last billboard is at i

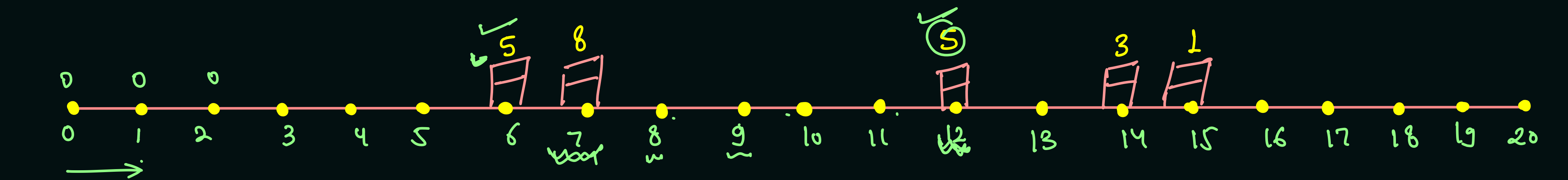
7 + 5 = 12

↳ Result = 11

Time Complexity → O(n²)

n ⇒ no. of billboards.

Method-2 Complexity → O(m) m → total distance of Highway



0	0	0	0	0	0	5	8	8	8	8	8	10	10	11	11 vs 5 11	11	11	11	11	11
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

i

Consider current billboard

Not consider current billboard

13 - 5 - 1

dp[i-t-1] + revenue[i] dp[i-1]

dp[i-t-1] + revenue[i] vs. dp[i-1]

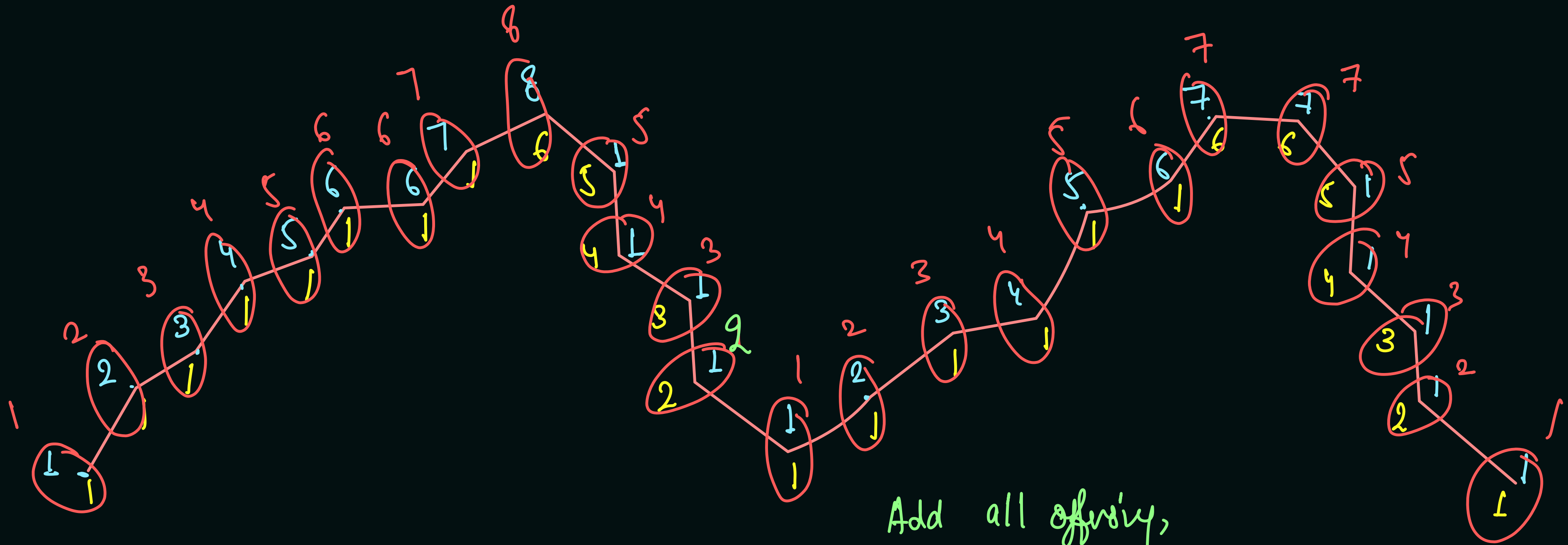
Result = dp[m] 11

temple Offerings :→

1. what?
2. How?
3. why?

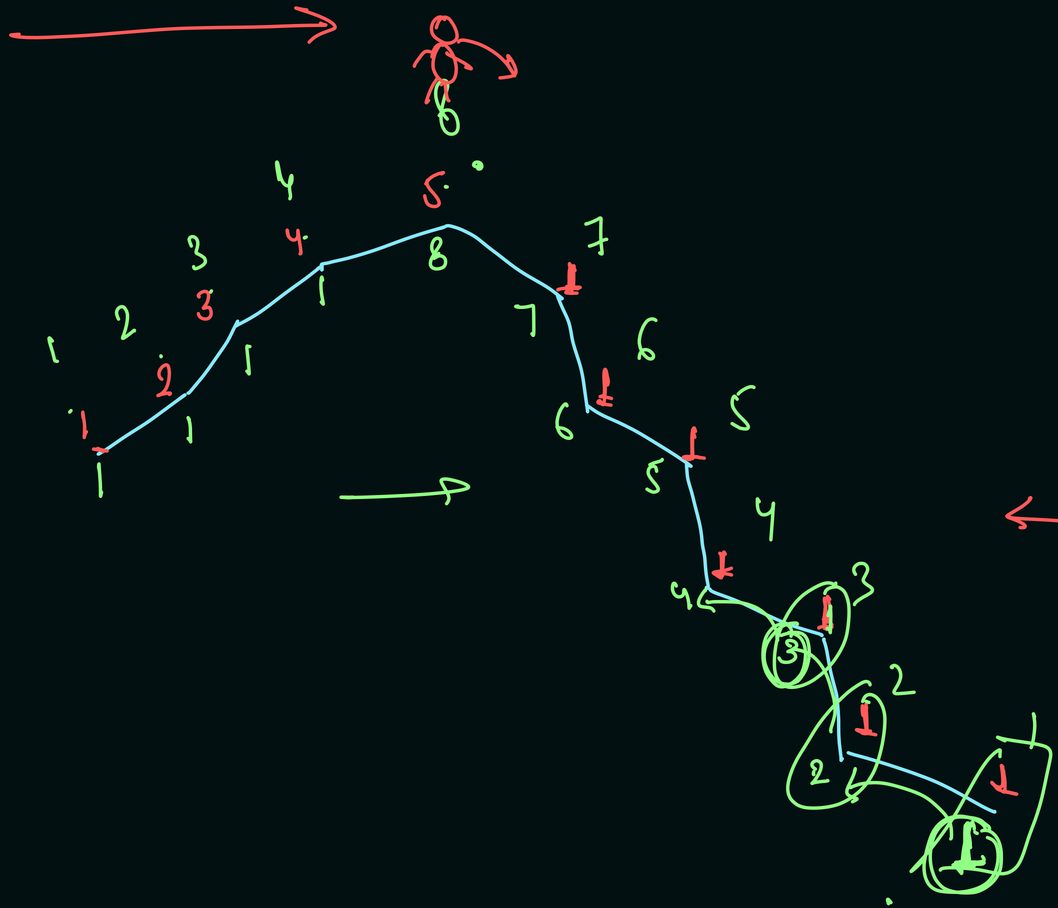
height t_2 > height t_1
offering t_2 > offering t_1

0 - offering not allowed,
min offering.



Add all offering,

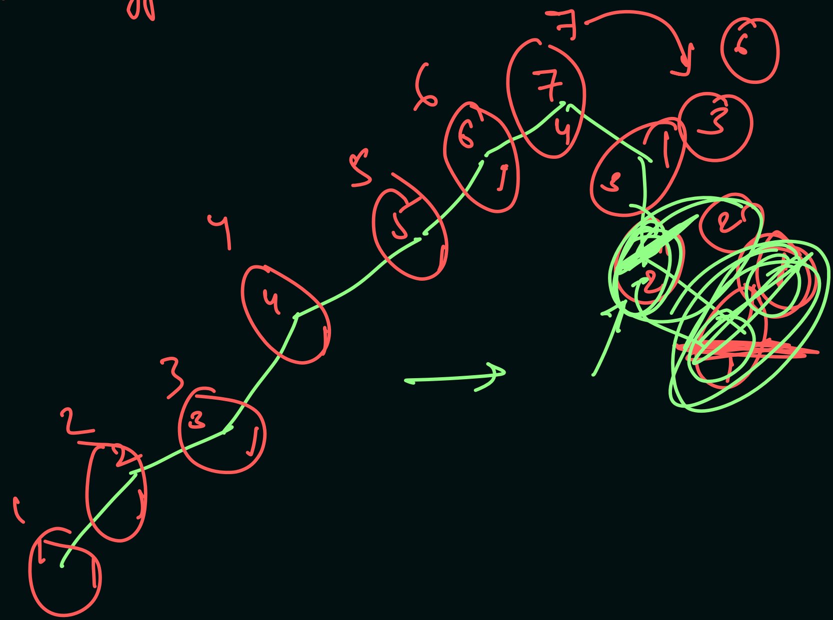
Steps
1. Begin with 1, and increment offering if ht[i] > ht[i-1].
2. if ht[i] < ht[i-1] then reset offering at 1.
3. Repeat same thing from Right to left
4. find max from both offerings & add it.] main



offering begin from 1

ht increase offering ↑

ht decr off ↓



Sum = 1 + 2 + 3