

### \* Problem statement :-

- We are climbing a stairs of  $n$  steps.
- Each time we climb, we can either take 1 step or 2 steps.
- Calculate the total distinct ways to climb to the top.

### \* Example :-

For  $n=1$ ,

O/P = 1, only 1 way possible.

For  $n=2$ ,

O/P = 2, Two possible ways:- 1+1 steps, 2 steps

For  $n=3$ ,

O/P = 3, 1+1+1, 1+2, 2+1

For  $n=4$

O/P = 5, 1+1+1+1, 1+1+2, 1+2+1, 2+1+1, 2+2

### \* Solution :-

$n=1 \rightarrow 1$

$n=2 \rightarrow 2$

$n=3 \rightarrow 3$

$n=4 \rightarrow 5$

} This is a fibonacci sequence with  $f_1 = 1$  &  $f_2 = 2$ .

### i) Approach 1 :- Binet's formula

- Directly apply it considering for  $(n+1)$  because our first term is starting from 1.

TC :-  $O(1)$

SC :-  $O(1)$

## 2) Approach 2:- Simple recursive

→ Results in TLE on LeetCode

→ Re-computation involved so avoid.

Note :- Recommended approach is Binet's formula.

Recursive approach can be optimized using DP.

Simple iterative approach works with  $TC = O(n)$  &  $SC = O(1)$

Explore matrix exponentiation approach as well.