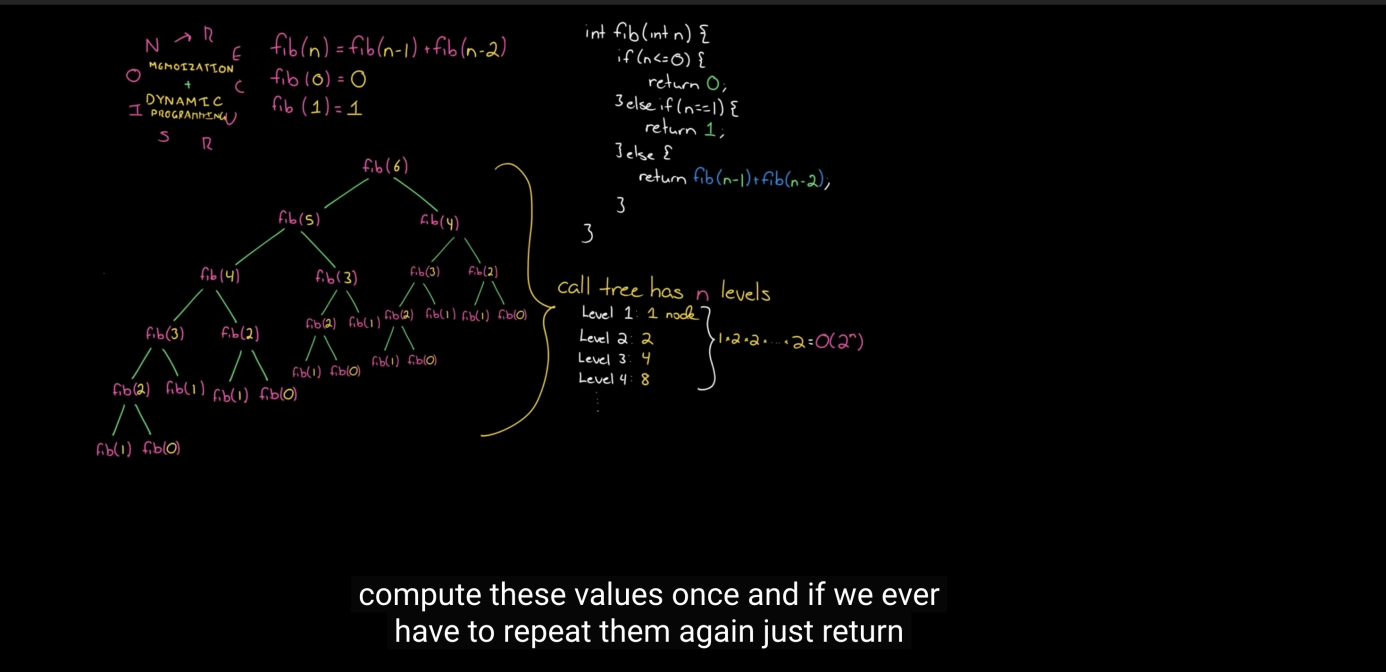
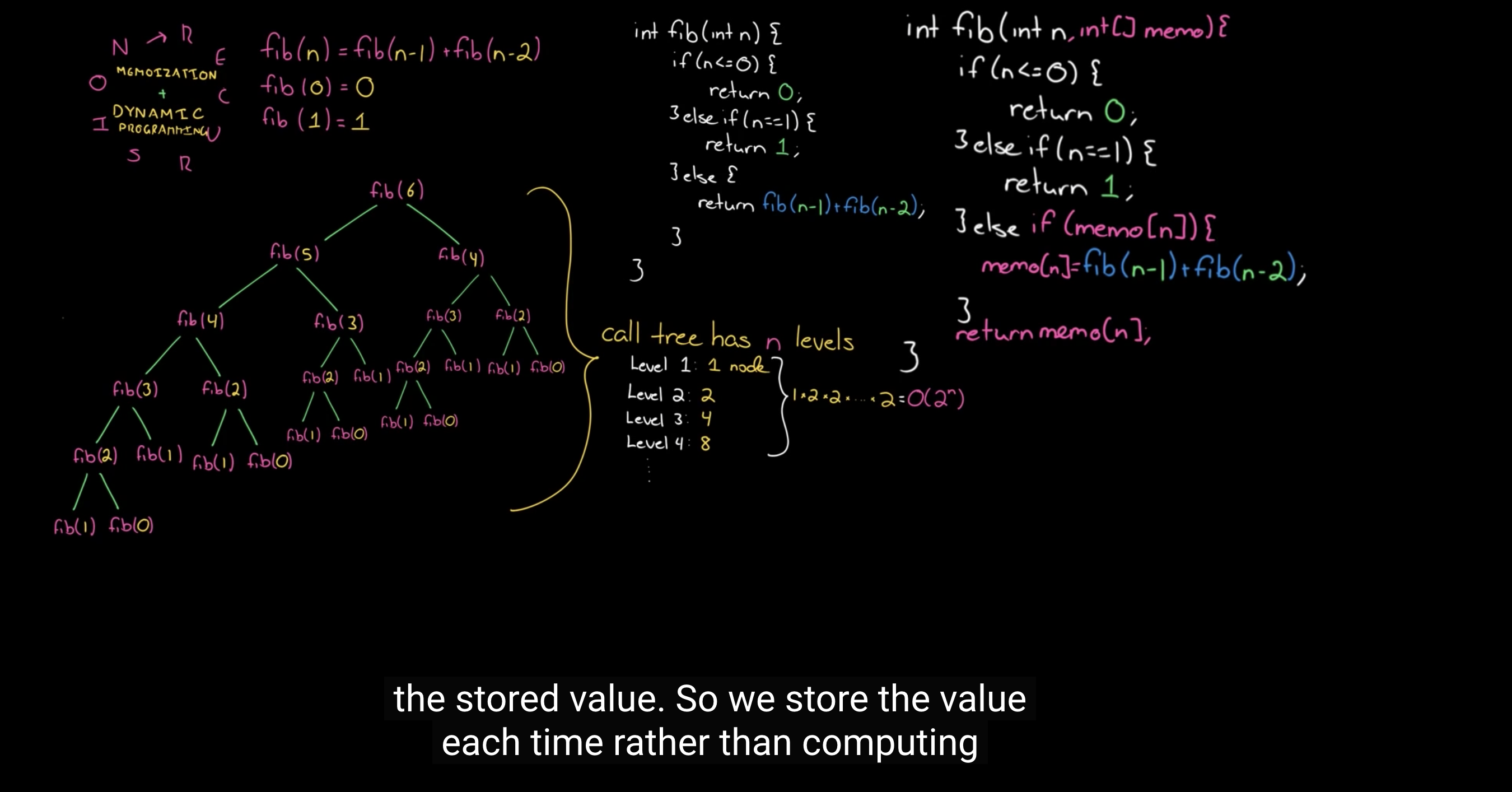
Reference: <https://www.youtube.com/watch?v=P8Xa2BitN3I>

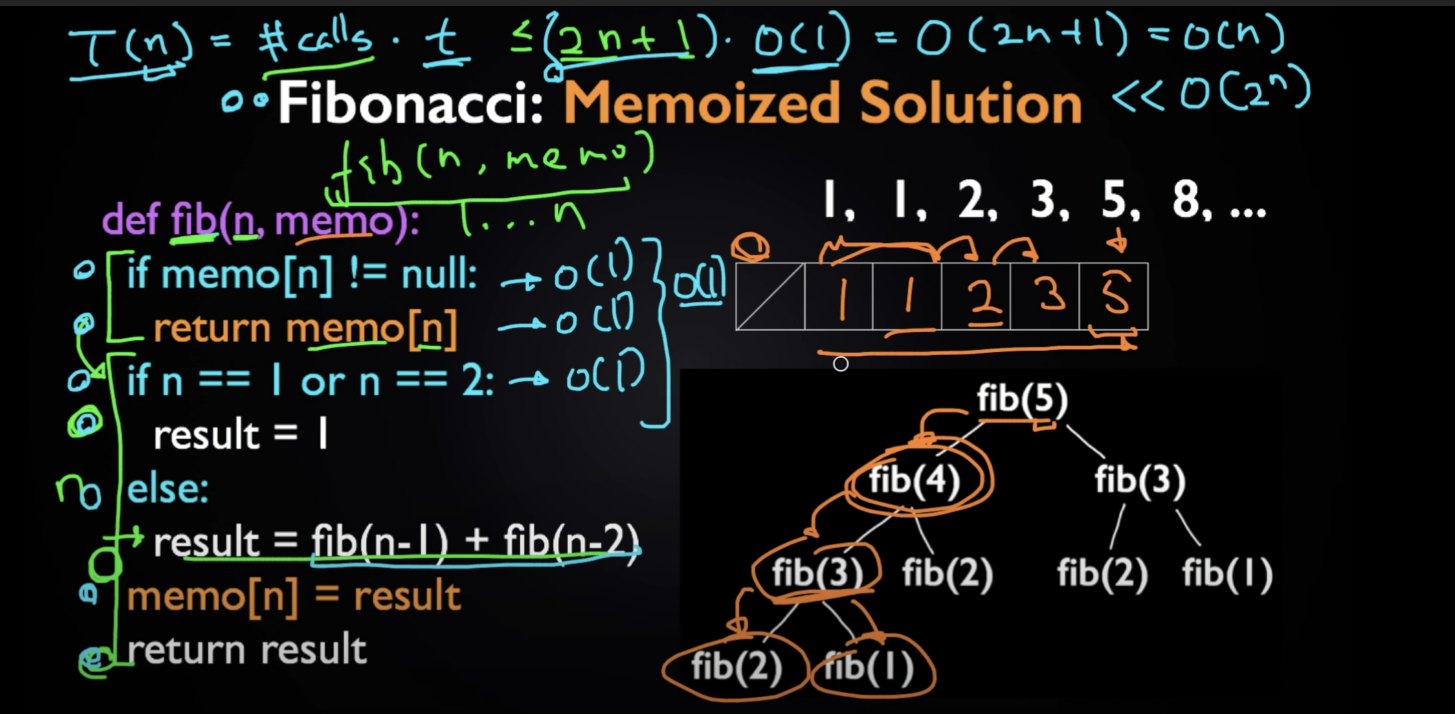
<https://www.youtube.com/watch?v=vYquumk4nWw&t=622s>

1. Fibonacci. Implemented by recursive method, it needs O(2n) time complexity

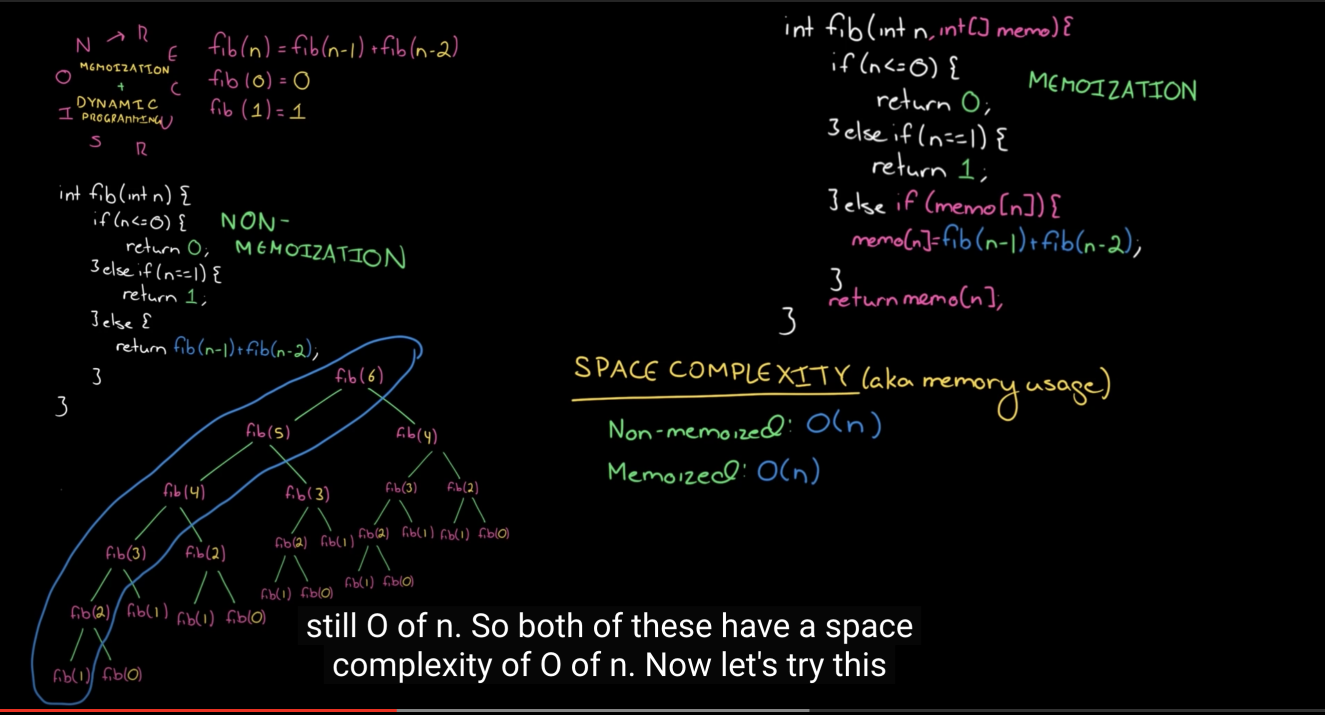


2. We store the value each time rather than computing it from scratch. - Memoization

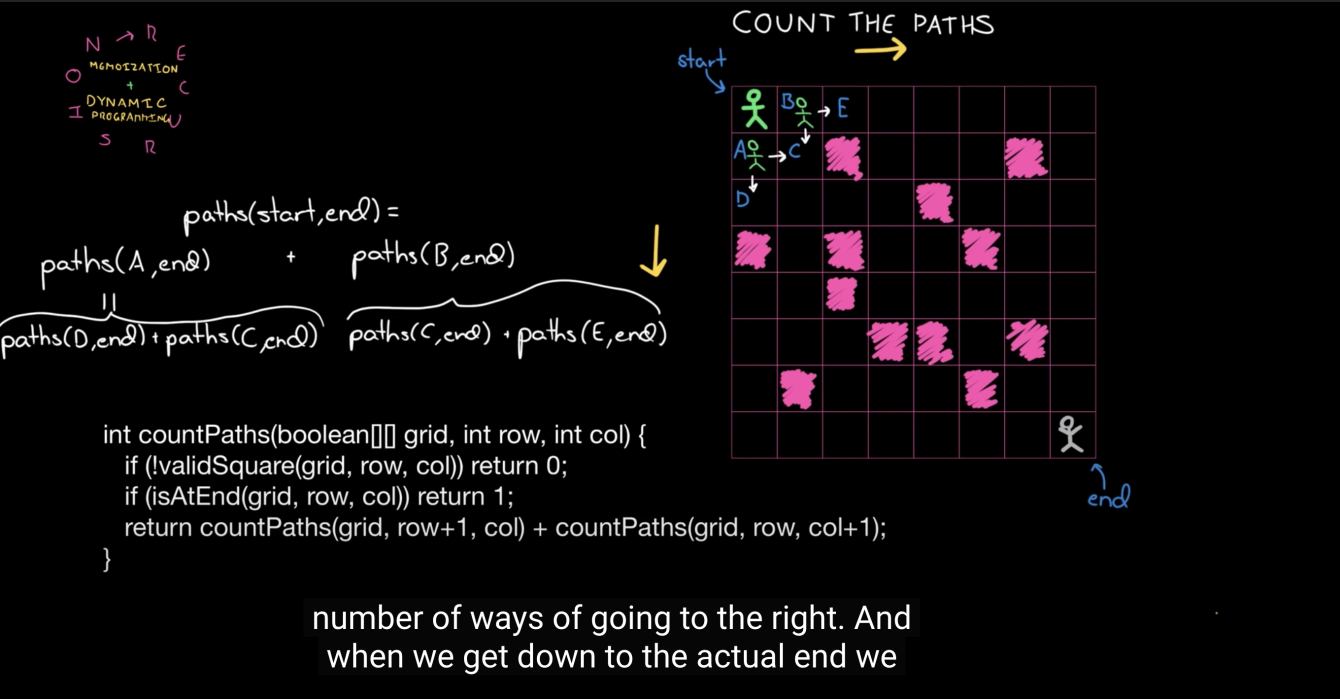


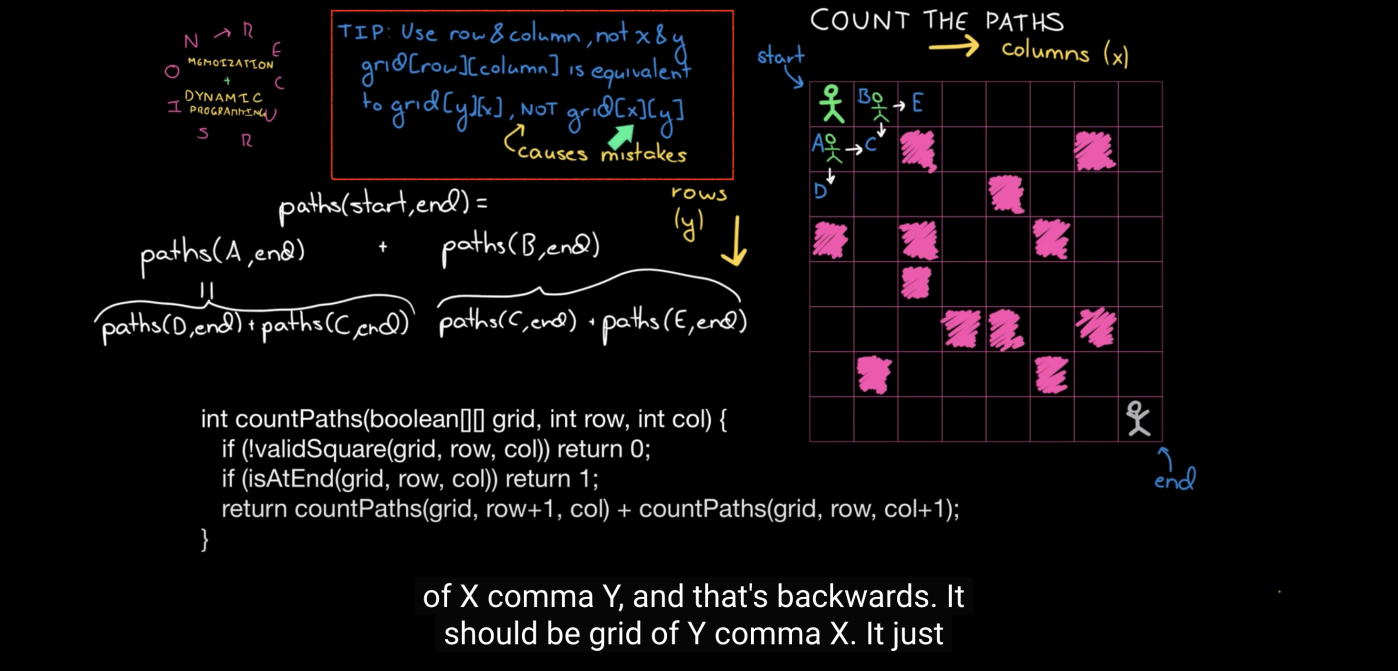


3. Space complexity

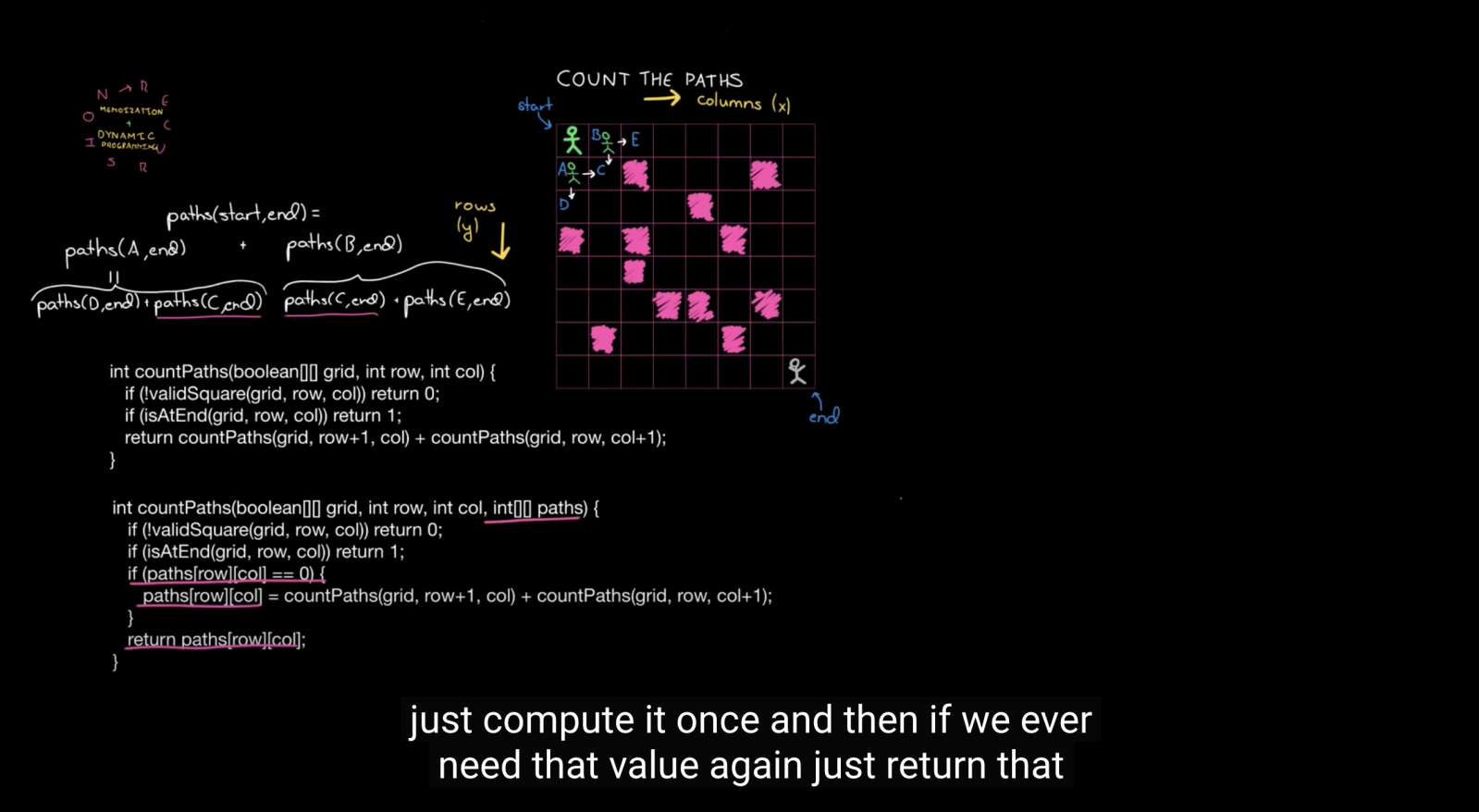


4. Count the number of paths from top-left to bottom-right. You can only move down or right.Use **recursive**.





5.Count the number of paths from top-left to bottom-right. You can only move down or right.Use **memoization**.



6.Bottom-up approach for Fibonacci

