**Recursion:-**

Recursion is a programming technique where a method calls itself to solve smaller parts of the same problem. It’s useful for breaking down problems that have a repetitive structure.

**Time Complexity:-**

The time complexity of this recursive method is **O(n)** where **n** is the number of years. It makes one recursive call per year.

**Optimization:-**

Recursion can lead to **repeated calls and stack overflow** for large n.

You can optimize it using:-

* **Memoization:**-
* Store already computed values in a dictionary or array to avoid recalculating them.
* Reduces time complexity from **exponential** to **linear** in many cases.
* **Iterative Approach-**
* Replace recursion with loops.
* Removes stack usage and improves performance, especially for large n.
* **Tail Recursion-** A special form of recursion that some compilers optimize internally to behave like iteration.

**Ayush Kumar (6363303)**