- **Discussing time complexity of recursive algorithm:**
  - In this recursive approach, the function works by calculating the future value one year at a time.
    It calls itself again and again for each previous year until it reaches year = 0 (base case).
  - So, if we're forecasting for n years, the function makes n recursive calls.
  - That means the time complexity of the algorithm is:
    O(n) — where n is the number of years we want to calculate for.

- **Explaining how to optimize recursive solution to avoid excessive computation:**

Using recursion for larger values of n can cause the program to make too many repeated calls and use more memory.

We can make it better in two ways:

1. Converting it to an iterative approach

Instead of recursion, we can simply use a for loop that calculates the value year by year.
This method is more efficient and doesn't use stack memory.
It still takes **O(n)** time, but runs faster and more smoothly in most cases.

2. Use memorization

In more complex versions where same values might get calculated again and again, we can store results for each year in memory.
That way, if the same year comes up again, we just reuse stored result saving time and avoiding extra calculations.