Financial Forecasting

1. Explain the concept of recursion and how it can simplify certain problems?

Ans. Recursion is a programming technique where a function calls itself to solve a problem. It simplifies problems by breaking them down into smaller, similar subproblems, ultimately leading to a solution.

Recursion embodies the divide-and-conquer strategy, breaking down a complex problem into smaller, more manageable subproblems. For problems with inherent recursive structures, recursive solutions can be more elegant and easier to understand than iterative solutions like problems involving hierarchical data structures like trees and graphs, where operations need to be performed at multiple levels.

1. Discuss the time complexity of your recursive algorithm?

Ans. In our algorithm, each call performs one multiplication and one recursive call. The total number of calls equals to the number of years so the time complexity of the algorithm is O(n) where n is the number of years.

1. Explain how to optimize the recursive solution to avoid excessive computation?

Ans. Each recursive call consumes stack memory. So it is necessary to optimise solution, although there are many methods such as Memoization, Iteration etc but in this case as there are no overlapping subproblems so Memoization will not be effective here. Here for small problems, we can replace recursion with iteration in order to remove stack space usage.