Algo-7

Financial Forecasting  
  
The provided Java program calculates the future value of an investment given its present value, growth rate, and the number of periods over which it will grow. Here's a detailed explanation of the code:

**Class Definition**

The class FinancialForecasting is defined to contain the methods for financial calculations.

**Method to Calculate Future Value**

The calculateFutureValue method takes three parameters:

* presentValue: The current value of the investment.
* growthRate: The rate at which the investment grows each period, expressed as a decimal (e.g., 0.05 for 5%).
* periods: The number of periods over which the investment grows.

The method initializes futureValue to the presentValue. It then iterates for the number of periods, updating futureValue by multiplying it by (1 + growthRate) in each iteration. Finally, it returns the computed futureValue.

**Main Method**

The main method does the following:

1. Creates a Scanner object to read input from the user.
2. Prompts the user to enter the presentValue, growthRate, and periods, reading each input from the console.
3. Calls the calculateFutureValue method with the user's inputs to compute the future value.
4. Prints the result to the console using formatted output.

**Time Complexity**

The time complexity of the calculateFutureValue method is O(n)O(n)O(n), where nnn is the number of periods. This is because the method contains a loop that iterates exactly periods times, performing a constant-time operation (multiplication and addition) in each iteration. Thus, the time complexity is linear with respect to the number of periods.

In the context of the entire program:

* Reading inputs from the user is O(1)O(1)O(1) for each input operation.
* Calculating the future value is O(n)O(n)O(n).
* Printing the result is O(1)O(1)O(1).

Therefore, the overall time complexity of the program is dominated by the O(n)O(n)O(n) complexity of the calculateFutureValue method.