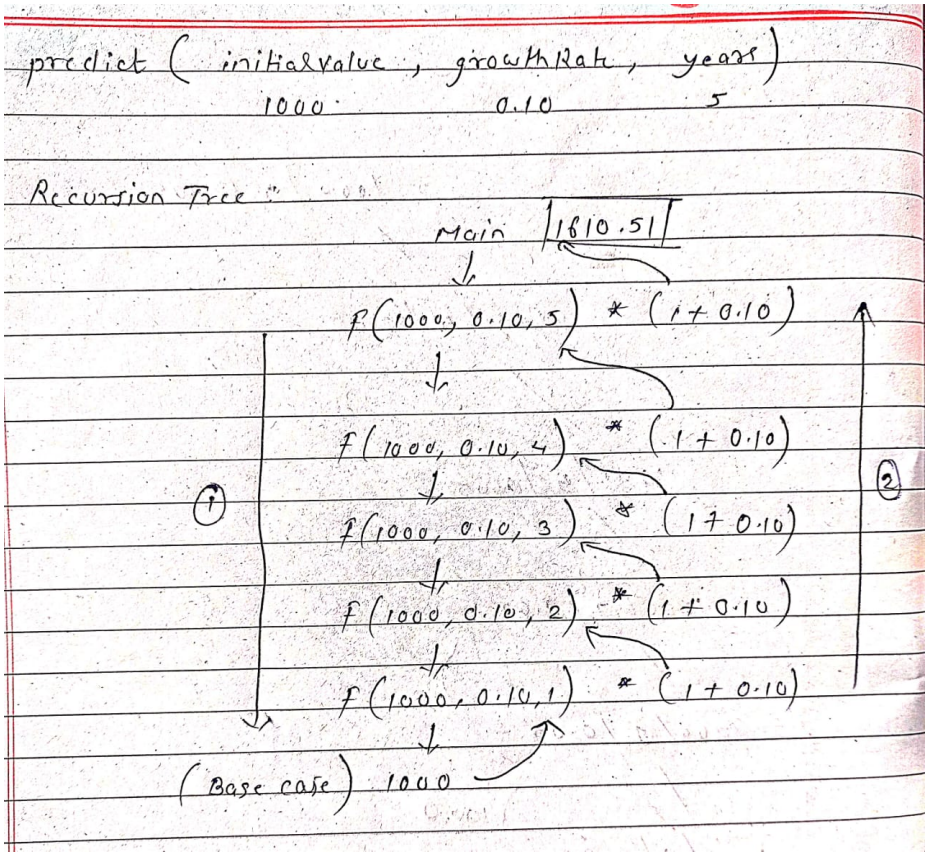
Recursion

- It is a technique where function calls itself until base case is reached.
- It helps in breaking down a bigger problem into smaller problems.
- E.g. we can simply calculate factorial of n by multiplying it with factorial of n-1
  do n * f(n-1) until n = 1

Time complexity

```
public static double predict(double initialAmount, double rate, int years) {
    if (years == 0) {
        return initialAmount;
    }
    double amount = predict(initialAmount, rate, years - 1);
    return  amount * (1 + rate);
}
```



Solution computes growth only once for each year taking linear time
Therefore, it takes O(n) time

Optimization

We can optimize overlapping recursive calls using dynamic programming

1. Memoization - involves storing the result of a function call so that If the same inputs occur again, we reuse the stored result avoiding the function call.
2. Tabulation - We solve the problem iteratively, starting from the base case. Involves storing result using bottom-up approach, where we use previously computed results to calculate the next one.