DYNAMIC PROGRAMMING

INTRODUCTION TO ALGORITHMS

Made by -
Shviam Joshi

INTRODUCTION

Dynamic programming (usually referred to as **DP**) is a very powerful technique to solve a particular class of problems. It demands very elegant formulation of the approach and simple thinking and the coding part is very easy. The idea is very simple, If you have solved a problem with the given input, then save the result for future reference, so as to avoid solving the same problem again.

There are two ways of doing this.

1.) Top-Down : Start solving the given problem by breaking it down. If you see that the problem has been solved already, then just return the saved answer. If it has not been solved, solve it and save the answer. This is usually easy to think of and very intuitive. This is referred to as ***Memoization***.

2.) Bottom-Up : Analyze the problem and see the order in which the sub-problems are solved and start solving from the trivial subproblem, up towards the given problem. In this process, it is guaranteed that the subproblems are solved before solving the problem. This is referred to as ***Dynamic Programming***.

Problem statement

Implement dp to find nth fibonacci numbers

The Fibonacci numbers are the numbers in the following integer sequence.

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ……..

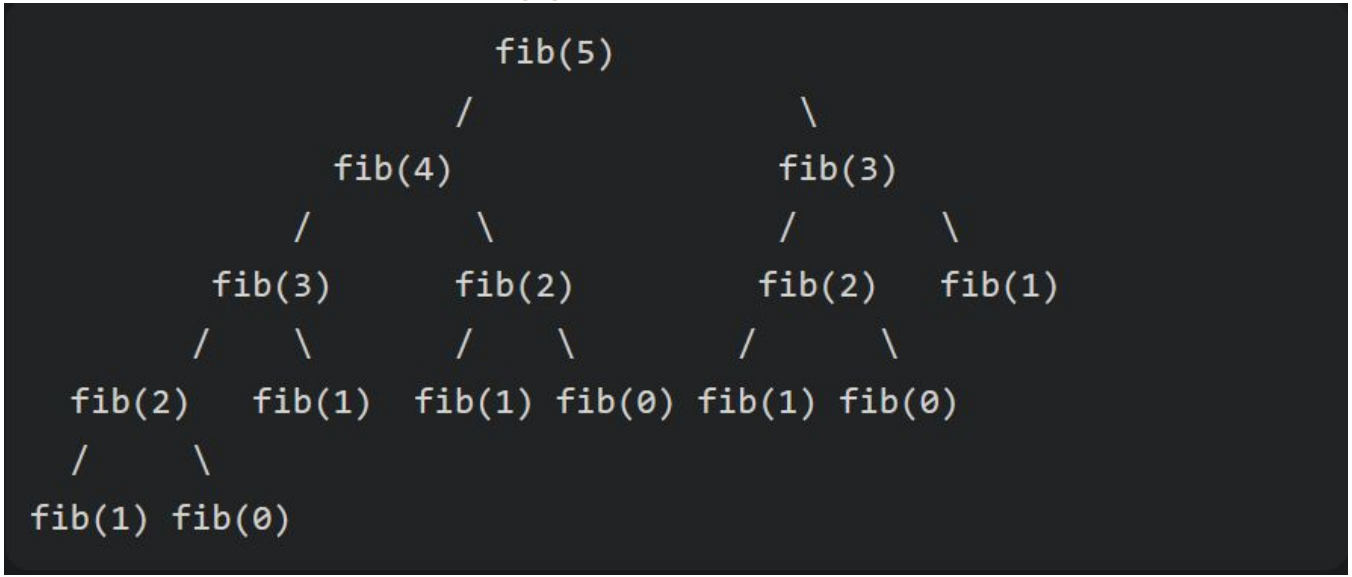
In mathematical terms, the sequence Fn of Fibonacci numbers is defined by the recurrence relation

$F_n = F_{n-1} + F_{n-2}$

$F_0 = 0$ and $F_1 = 1$.

Approach

A lot of extra work in the naive approach



We can avoid the repeated work done is the method 1 by storing the Fibonacci numbers calculated so far.

```python
def fibonacci(n):

    # Taking 1st two fibonacci nubers as 0 and 1
    FibArray = [0, 1]

    while len(FibArray) < n + 1:
        FibArray.append(0)

    if n <= 1:
        return n
    else:
        if FibArray[n - 1] == 0:
            FibArray[n - 1] = fibonacci(n - 1)

        if FibArray[n - 2] == 0:
            FibArray[n - 2] = fibonacci(n - 2)

    FibArray[n] = FibArray[n - 2] + FibArray[n - 1]
    return FibArray[n]

print(fibonacci(9))
```