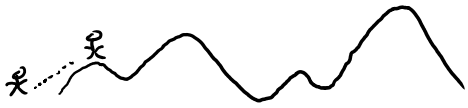
2019-03-25 Dynamic Programming

Monday, March 25, 2019 2:59 PM

- A greedy algorithm can result in a suboptimal solution
  - They get trapped in "local maxima"

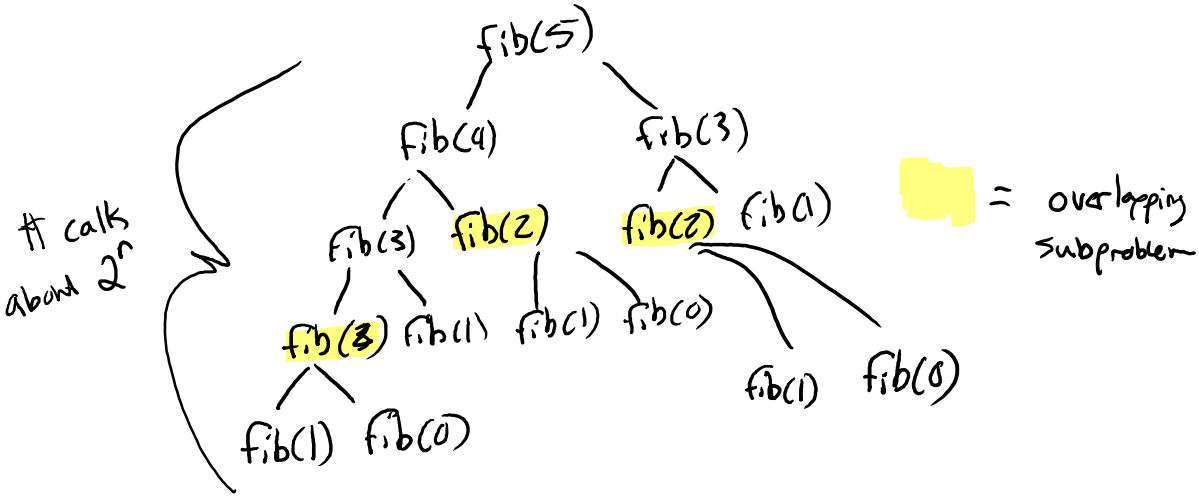


- Where local maxima exist, a greedy approach will not work, must revert back to brute force. SLOW!
- Dynamic programming is technique that can be applied to some brute force approaches when:
  - The sub solution of a problem is useful to a later solution
  - Subproblems overlap

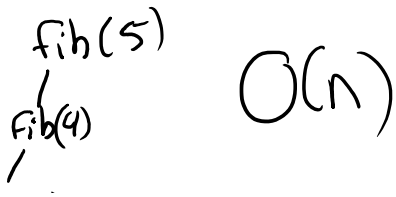
Consider the recursive Fibonacci solution:

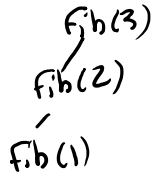
```
int fib(n)
{
  if n == 0 return 0;
  if n == 1 return 1;
  return fib(n - 1) + fib(n - 2);
}
```

Consider the recursion tree for fib(5)



- When subproblems overlap and solutions inform future work, dynamic programming can be used to radically speed up an algorithm. All dynamic programming does is use a lookup table to store and retrieve prior answers. This is officially called memoization.
- The same recursion tree using memoization:





- Using a lookup table is called "top down" dynamic programming. Top-down starts with the answer that you want to achieve and works to a base case.
    - Thus, top-down is recursive
- Using "bottom-up" dynamic programming, you start at the base case and work to the desired answer.
    - Bottom-up dynamic programming is iterative
- Asymptotically, top-down and bottom-up are the same time complexity
- Practically, bottom-up will almost always be faster

Rod Cutting Problem

- Variation of the 0..1 knapsack problem
- Given some rod of length N, how can we cut the rod in order to maximize profit?
- Theme: Humboldt Lumber Mill
    - Trees come in all lengths. How do we cut the tree in order to make the most money?
    - Payout schedule
    -

Feet	6	8	10	12	16	18	20	24
Payout	2	3	4	5	7	9	10	12

- Greedy approach: always cut using the most profitable payout.
    - How much money would greedy make on a 36 foot board?
        - 36 - 24 = 12 - 12 = 0
        - 12 + 5 = 17
    - True answer: 2x 18 foot cuts ($18)
- Dynamic approach: compute every possible combination and remember results.