# Dynamic programming

Lesson in recycling

# Classic use cases

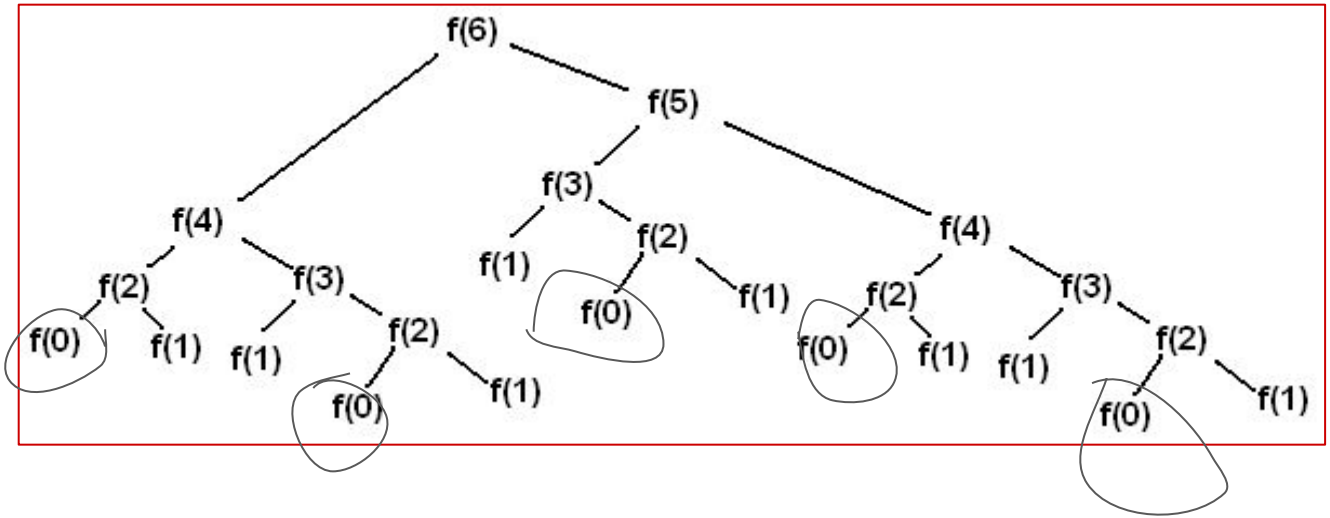
- Can split a problem into smaller sub problems
- Reuse

# Methods to solve

- Intuitive way of pattern recognition
- Recursion
  - Calling a person to perform the same task for you once you have done your part
  - How do you identify which row you are sitting in a stadium?
  - You pass the news that you want the row number to all the rows below you and the person in the first row returns saying row number “1” and all others return the row number by adding “1” to it, all the way up to you.
- Memoization where store the values calculated
  - Some cases in recursion where values can be stored
- Tabulation
  - Calculate from bottom up

# Fibonacci series

- 0 1 1 2 3 5 8
-



# Fibonacci

- Problem can be split into sub problems
- Memoization
  - Cache the F(0) and F(1) - do not calculate it every time
- Tabulation
  - Calculate F(0) and F(1) and store it and calculate it upwards

# Pascal’s triangle

- Code for all the approaches
- Discuss

# Now solve Fibonacci!

- Try all the approaches

# Other classic problems

- Longest common subsequence “ABCDGH” and “AEDFHR” is “ADH”
  - Used in finding the difference between two files
- Knapsack problem: Consider maximum capacity for a bag and you have weights for N items and their corresponding Values. The values need to be maximized
  - Used by download managers. Data is broken into chunks and value of the download is maximized
- Coin change problem: Get 5 Rs with 1/2/3/4 Rs coins. A common problem
- Many more! Read up!