L1: What is Dynamic Programming

# Define & Demonstrate need

* Consider Dynamic Programming as a black box that takes in a problem statement as an input such that it produces a solution as an output.
* It helps solve problems in polynomial time where naïve approach would take exponential time. O(NC) [Polynomial time] and O(CN) [Exponential time]
* It is an optimization technique, to go from exponential to linear complexity

# Properties of DP problems

* **Optimal Substructure:** A problem has optimal substructure property when the optimal solution of the problem can be constructed from the optimal solution of the substructures of the given problem.
  + **Where,** 
    - **Optimal** means best or most favorable
    - **Substructure** sub problem of the main problem
  + Consider a problem X, we know that a complex problem can be systematically solved by breaking it down. Consider the fragment solved first to be X­‑1, X2, X3 … Xn.
  + In the above case we can first solve X1, then use the results of X1 to solve for X2 sequentially without resolving X1 hence reducing the complexity.
* **Overlapping Sub problems:** When you break a problem into sub problems into sub problems you will notice that you need to re-calculate some work multiple times. Consider the Fibonacci example, in which the colored boxes in the following diagram highlight recalculation of a given sub problem multiple times.