**Summations:**

**Logs:**

**Asymptotic Notation:**

**Recurrence Relations:**A **recurrence relation** is an equation or inequality that describes a function in terms of its value on smaller inputs. They’re useful for analyzing the running **time** of recursive algorithms.

**Master Method:**

Case 1

implies

**Dynamic Programming:**A **recurrence formula** describes the **strategy** of an algorithm.

**Fibonacci**:

**Product sum**:

**Change-making:** (time), (space)   
**Backtracking:**

**Greedy Algorithms:**Make the **best choice available** during each iteration and **don’t look back**.

Finite Geometric Progression:

Rankings

Common Recurrences

where:  
 is a positive polynomial function

Case 2

implies

Requires **Optimal Substructure** and **Overlapping Solutions**.

**LCS**:   
**Optimal LCS**:

**U. Knapsack**:

**0-1 Knapsack**:

Used to solve problems where we want to **find all possible solutions**.

General Steps:   
1) If in final state, do bookkeeping and return  
2) Loop through all possible choices:  
 a. Make a choice and check constraints  
 b. Recurse to smaller problem  
 c. Unmake choice  
  
Benefits:  
Easy to implement and efficient.  
Limitations:  
Does **not** always return optimal solutions.  
Hard to design. Difficult to verify.

Infinite Geometric Progression:

Substitution Method – substitute smaller values of the recurrence relation back into the original equation until a pattern emerges. Use base case to solve for the general formula.

and are constants

Case 3

implies

**Bottom-up** vs **Top-down**. **Memoization**.

Steps: 1) Identify parameters, 2) Identify subproblem, 3) Define recursive formula, 4) Implement naïve recursive solution, 5) Turn recursive formulation into DP algorithm

**Rod Cutting:** naïve=, DP=

Permutations:   
PowerSet:   
N-Queens:   
Combination Sum w Repetition:   
Combination Sum w/o Repetition:

Keys: 1) Choice 2) Constraint 3) Goal

Requires **Greedy Choice Property** (locally optimal leads to globally optimal) and **Optimal Substructure**.