**DYNAMIC PROGRAMMING (DP)**

**KNAPSACK FORMAT :**

1. **0-1 Knapsack Problem:**
   1. **Recursive Code : <https://ide.geeksforgeeks.org/2HViNZHe96>**
   2. **Memorization Code : <https://ide.geeksforgeeks.org/Z8LJXZ6AKr>**
   3. **Top-Down Code : <https://ide.geeksforgeeks.org/rBfBxr2J2G>**
2. **0-1 Knapsack Similar Problems**
   1. **Subset Sum :** 
      1. **Recursive Code : <https://ide.geeksforgeeks.org/Jx2y3dCQAx>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/Tf5mVOqJLC>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/ckqnaWy8Ko>**
   2. **Equal Sum Partition :** 
      1. **Recursive Code : <https://ide.geeksforgeeks.org/ZXVAO2027B>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/P4K9kqng5s>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/ZV9IcrjGk5>**
   3. **Count of Subsets of a given Sum :** 
      1. **Recursive Code : <https://ide.geeksforgeeks.org/VBiCNZGQ8B>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/GM6duydxO8>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/2QKTReCv96>**
   4. **Minimum Subset Sum Difference :**
      1. **Recursive Code : <https://ide.geeksforgeeks.org/Jx0rPszy65>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/h37VHxlwfh>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/OykjBPzl1J>**
   5. **Count no of subsets of given Difference :**
      1. **Recursive Code : <https://ide.geeksforgeeks.org/f7XH4r8Bg0>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/xkNQ0Mxtf8>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/WP5i6qtjnX>**
   6. **Target Sum :**
      1. **Recursive Code : <https://ide.geeksforgeeks.org/e9lSDtCONt>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/Z2OfZA1DTZ>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/acLzUGe8rr>**

**UNBOUNDED KNAPSACK FORMAT :**

1. **Unbounded Knapsack Problem :**
   1. **Recursive Code : <https://ide.geeksforgeeks.org/f0GpuDFj5X>**
   2. **Memorization Code : <https://ide.geeksforgeeks.org/nereuTm1L2>**
   3. **Top-Down Code : <https://ide.geeksforgeeks.org/tXZ1Wlc8nC>**
2. **Unbounded Knapsack Similar Problems**
   1. **Rod Cutting Problem :**
      1. **Recursive Code : <https://ide.geeksforgeeks.org/r5l3TWTNJA>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/sqXLUYpqKS>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/o3Pu5djlmK>**
   2. **Coin Change - I (Max # of ways) :**
      1. **Recursive Code : <https://ide.geeksforgeeks.org/E26qqJJZ7m>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/TvRVIMEdt1>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/lZSP17sZz8>**
   3. **Coin Changer - II (Min # of coins) :**
      1. **Recursive Code : <https://ide.geeksforgeeks.org/NjJGUn0o8x>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/2RP3tyrqSW>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/8QPqvR9ZrS>**
   4. **Maximize Cuts Segment (Max # of Cuts-->{Coins})**Given an integer ****N**** denoting the Length of a line segment. You need to cut the line segment in such a way that the cut length of a line segment each time is either ****x**** , ****y**** or ****z****. Here x, y, and z are integers.  
      After performing all the cut operations, your****total number of cut segments must be maximum****.  
        
      You only need to complete the function ****maximizeTheCuts()****that takes n, x, y, z as parameters and returns ****max number cuts****.  
        
      ****Expected Time Complexity**** : O(N)  
      ****Expected Auxiliary Space****: O(N)  
        
      ****Constraints****1 <= N, x, y, z <= 10^4
      1. **Recursive Code : <https://ide.geeksforgeeks.org/0OBP4YMlDU>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/DdN09zyvrv>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/OCg6sRoDcd>**

**LONGEST COMMON SUBSEQUENCE(LCS) FORMAT :**

1. **Longest Common Subsequence (LCS)**
   1. **Recursive Code : <https://ide.geeksforgeeks.org/78UutYGCJ1>**
   2. **Memorization Code : <https://ide.geeksforgeeks.org/SiUhiJHxMv>**
   3. **Top-Down Code : <https://ide.geeksforgeeks.org/WRq3dM4ie5>**
2. **Longest Common Subsequence (LCS) similar Problems :**
   1. **Longest Common Substring :** 
      1. **Recursive Code : <https://ide.geeksforgeeks.org/jMVtdaRhMb>**
      2. **Memorization Code :**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/ZjdfksQrMJ>**
   2. **Print LCS b/w 2 Strings :**
      1. **Recursive Code :**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/XOowSYRN8c>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/ngeiKIfqvO>**
   3. **Shortest Common Supersequence :**
      1. **Recursive Code : <https://ide.geeksforgeeks.org/rsk6jRtstf>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/QKu1HYLEGr>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/zH9a9JwK2l>**
   4. **Minimum # of insertion & deletion :**
      1. **Recursive Code : <https://ide.geeksforgeeks.org/Ng9pctguJf>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/4CNljOH9YL>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/JAgBnfltSz>**
   5. **Largest Palindromic Subsequence :**
      1. **Recursive Code : <https://ide.geeksforgeeks.org/dq7O4RN7Qh>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/X0I4IxH7rk>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/otQb1lq4pk>**
   6. **Minimum # of deletion in a String to make it a Palindrome :**
      1. **Recursive Code : <https://ide.geeksforgeeks.org/U2PzFEwtgu>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/ew44bEs5v6>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/zL6blbQhOe>**
   7. **Print Shortest Common Supersequence(SCS) :**
      1. **Recursive Code :**
      2. **Memorization Code :**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/W6ljJ1lmst>**
   8. **Largest Repeating Subsequence(LRS) :** 
      1. **Recursive Code : <https://ide.geeksforgeeks.org/ecw1Ty0WPB>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/I7MSXobvE1>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/RIu2mW0rWB>**
   9. **Sequence Pattern Matching :**
      1. **Recursive Code : <https://ide.geeksforgeeks.org/nrv8yFFJcm>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/rDlwPsOymH>**
      3. **Top-Down Code : <https://ide.geeksforgeeks.org/vIWZApeMGg>**
   10. **Minimum # of insertions in a String to make it a Palindrome :**
       1. **Recursive Code : <https://ide.geeksforgeeks.org/Q9LgQEw1Qz>**
       2. **Memorization Code : <https://ide.geeksforgeeks.org/SqcjujaqRx>**
       3. **Top-Down Code : <https://ide.geeksforgeeks.org/bF0K33TdrT>**

**MATRIX CHAIN MULTIPLICATION(MCM) FORMAT :**

1. **Matrix Chain Multiplication(MCM) :** 
   1. **Recursive Code : <https://ide.geeksforgeeks.org/pSEJupVTfJ>**
   2. **Memorization Code : <https://ide.geeksforgeeks.org/JLgu9xWGYQ>**
   3. **Top-Down Code :**
2. **Matrix Chain Multiplication Related Problems :** 
   1. **Palindromic Partitioning :**
      1. **Recursive Code : <https://ide.geeksforgeeks.org/qgrmeGJSxw>**
      2. **Memorization Code : <https://ide.geeksforgeeks.org/CoiGrLP9mB>**
      3. **Memorization Optimized : <https://ide.geeksforgeeks.org/Q4jEM65TWe>**
      4. **Memorization Super Optimized : <https://ide.geeksforgeeks.org/0aMjrXrPdb>**
      5. **Top-Down Code :**