

DP String Problem Variants



what are variant of string question on dp



Dynamic Programming (DP) problems involving strings come in a variety of types and difficulty levels. Below are some common **variants of string problems** in DP, grouped by their themes:

1. Subsequence Problems

- **Longest Common Subsequence (LCS)**
 - Find the LCS between two strings.
- **Shortest Common Supersequence (SCS)**
 - Find the shortest string that has both strings as subsequences.
- **Number of Subsequences**
 - Count the number of ways one string can appear as a subsequence in another.
- **Longest Palindromic Subsequence**
 - Find the longest subsequence of a string that is a palindrome.
- **Count Palindromic Subsequences**
 - Count the total number of palindromic subsequences in a string.

2. Substring Problems

- **Longest Common Substring**
 - Find the longest substring common to two strings.
- **Longest Palindromic Substring**
 - Find the longest substring that is a palindrome.
- **Distinct Substrings**
 - Count the number of distinct substrings in a string.
- **Count of Substrings with Certain Properties**
 - For example, substrings with at most `k` distinct characters.

3. Edit Distance Problems

- **Edit Distance (Levenshtein Distance)**
 - Find the minimum number of operations (insertions, deletions, or substitutions) to convert one string to another.
- **Minimum Insertions to Make a String Palindrome**
- **Minimum Deletions to Make a String Palindrome**

4. Partitioning Problems

- **Palindrome Partitioning**

- Partition the string into the minimum number of palindromic substrings.
 - **Partitioning into K Palindromic Substrings**
 - Determine whether a string can be partitioned into exactly `k` palindromic substrings.
-

5. Matching Problems

- **Wildcard Pattern Matching**
 - Match a string with a pattern containing `*` and `?`.
 - **Regular Expression Matching**
 - Match a string with a pattern involving regular expressions.
 - **String Alignment Problems**
 - Align two strings using scoring systems (useful in bioinformatics).
-

6. Counting Specific Strings

- **Count Distinct Subsequences**
 - Count distinct subsequences of a string.
 - **Count Strings that Match a Pattern**
 - Count the number of subsequences matching a specific pattern.
 - **Count Binary Strings Without Consecutive 1s**
 - Specific string formation problems based on given rules.
-

7. Optimization Problems

- **Maximum Sum of Non-Overlapping Substrings**
 - Find the maximum sum of values assigned to substrings without overlap.
 - **Optimal Parenthesization of Strings**
 - E.g., evaluating the minimum cost of matrix chain multiplication using strings.
-

8. Miscellaneous

- **String Interleaving**
 - Check if a string is an interleaving of two other strings.
 - **Decode Ways**
 - Given a string of digits, count the number of ways to decode it.
 - **Word Break Problem**
 - Determine if a string can be segmented into valid dictionary words.
 - **Burst Balloons (variation using strings)**
 - Analogous problems with strings instead of arrays.
-

Practice Tips:

1. Start with classical problems like LCS, LPS, and Edit Distance.
2. Gradually move to more challenging ones like Word Break and Regular Expression Matching.
3. Practice variations and edge cases to deepen understanding.

Let me know if you'd like explanations or code examples for specific problems!