1. **Introduction to Regular Expressions**

Regular Expressions (**Regex**) are sequences of characters that define a search pattern. They are widely used in: **String matching,** **Text processing**, **Pattern recognition.**

**Common Regex Symbols**

| **Symbol** | **Meaning** |
| --- | --- |
| **.** | Matches any single character. |
| **\*** | Matches zero or more of the preceding character. |
| **+** | Matches one or more of the preceding character. |
| **?** | Matches zero or one of the preceding character. |
| **[]** | Matches any one character inside the brackets. |

1. **What is Dynamic Programming (DP)?**

**Dynamic Programming (DP)** is an **optimization technique** used to solve problems by **breaking them down into smaller overlapping subproblems** and storing their results to avoid redundant computations.

Instead of solving the same subproblem multiple times (as in recursion), **DP stores the results** (memoization) or builds solutions iteratively (bottom-up approach).

**Why Use Dynamic Programming?**

1. **Avoids Redundant Computation**
   * Recursion may recompute the same subproblem multiple times, leading to **exponential time complexity (O(2^n))**.
   * DP **stores results** to prevent recomputation, reducing complexity to **polynomial time (O(n^2) or O(n))**.
2. **Optimizes Recursive Solutions**
   * **Memoization (Top-Down DP)**: Uses recursion + caching.
   * **Tabulation (Bottom-Up DP)**: Uses iterative table-filling.