

## **Striver DP Series : Dynamic Programming Problems → TUF**

Dynamic Programming can be described as storing answers to various sub-problems to be used later whenever required to solve the main problem.

The two common dynamic programming approaches are:

Memoization: Known as the “top-down” dynamic programming, usually the problem is solved in the direction of the main problem to the base cases.

Tabulation: Known as the “bottom-up ” dynamic programming, usually the problem is solved in the direction of solving the base cases to the main problem.

This post contains some hand-picked questions by Striver to learn or master Dynamic Programming. The post contains popular dynamic programming problems along with detailed tutorials (both text and video). You can also practice the problem on the given link before jumping straight to the solution.

### **1 → DP solution [ Using Recursion ]**

- Express Everything in terms of index
- Do all the stuff on that index
- According to question [find min, max, sum-up, longest, shortest, total number of ways]

### **2 → DP solution [memorization]**

- Identification of overlapping problems
- Reduce the overlapping recursive calls
- Create Array/Matrix using changeable variables
- Store the recursive calls
- Re-use the recursive calls

### **3 → DP solution [Tabulation]**

- Create matrix/Array using changeable variables
- convert the base condition into initialization of DP array/Matrix
- Copy the recurrence relation [fill using iterative way]

### **4 → DP solution [ Optimizing Space complexity ]**

- If you don't know Space Optimization, you don't know DP.
- According to the question, do you really need an array or matrix?
- Either You can just need Last Two value
- prev\_value1 and prev\_value2, to Calculate current\_value
- Inorder to remove the extra space complexity
- we can use DP on space Optimization.