BOSSCODER
ACADEMY

MASTER DYNAMIC PROGRAMMING in DSA

10 DAYS ROADMAP

PROGRAMMING

Pre-Requisite

Basics and Recursion

Goal

Understand the basics of algorithms and data structures, and practice solving problems using recursion.

Topics to cover

- Arrays, linked lists, trees, graphs (if unfamiliar, spend some time on each)
- Recursion and recursive calls

Resources

- Read about arrays, linked lists, trees, and graphs on your preferred learning platform or textbook.
- Recommended resources:
  - GeeksforGeeks - Data Structures
  - Introduction to Algorithms - MIT OpenCourseWare

Practice Questions

1. Write a recursive function to calculate the factorial of a number.
2. Implement a recursive function to reverse a linked list.
3. Solve the Tower of Hanoi problem using recursion.
4. Find the nth term of the Fibonacci sequence using recursion.
5. Implement a recursive solution for the binary search algorithm.

Day 1

Principles of Dynamic Programming

Goal

Learn the principles behind dynamic programming and understand optimal substructure.

Topics to cover

- Dynamic programming principles
- Optimal substructure

Resources

- Watch the lecture series on dynamic programming by MIT OpenCourseWare:
  - MIT 6.006 Introduction to Algorithms, Lecture 15: Dynamic Programming I
  - MIT 6.006 Introduction to Algorithms, Lecture 16: Dynamic Programming II
- Recommended resources:
  - TopCoder - Dynamic Programming
  - GeeksforGeeks - Dynamic Programming

Practice Questions

1. Find the number of unique paths in a grid using dynamic programming.
2. Solve the Coin Change problem using dynamic programming.
3. Implement a dynamic programming solution for the Longest Increasing Subsequence problem.
4. Find the maximum sum of a subarray using Kadane's algorithm.

Day 2

Overlapping Subproblems and Memoization

Goal

Gain a clear understanding of overlapping subproblems and how to use memoization.

Topics to cover

- Overlapping subproblems
- Memoization (top-down approach)

Resources

- Read the article on memoization by GeeksforGeeks:
  - GeeksforGeeks - Memoization
- Recommended resources:
  - Dynamic Programming: From Novice to Advanced - TopCoder
  - Dynamic Programming - CodeChef

Practice Questions

1. Solve the Longest Common Subsequence problem using memoization.
2. Implement a memoized solution for the 0/1 Knapsack problem.
3. Find the minimum number of coins required to make a given sum using memoization.
4. Solve the Rod Cutting problem using memoization.

Day 3

Tabulation and Dynamic Programming Techniques

Goal

Learn the tabulation (bottom-up) approach and explore additional dynamic programming techniques.

Topics to cover

- Tabulation (bottom-up approach)
- Dynamic programming techniques (e.g., prefix sums, state compression, bitmasks)

Resources

- Read about tabulation and its advantages over memoization on GeeksforGeeks:
  - GeeksforGeeks - Tabulation vs Memoization
- Explore additional dynamic programming techniques on Codeforces:
  - Codeforces - Dynamic Programming

Practice Questions

1. Solve the Fibonacci sequence problem using a tabulated approach.
2. Implement a bottom-up solution for the Longest Increasing Subsequence problem.
3. Find the number of ways to reach the top of a staircase using tabulation.
4. Solve the Subset Sum problem using dynamic programming techniques.
5. Implement the 0/1 Knapsack problem using a bottom-up approach.

Day 4

Classic Dynamic Programming Problems

Goal

Solve classic dynamic programming problems to reinforce your understanding.

Topics to cover

- Classic dynamic programming problems

Resources

- Solve dynamic programming problems on LeetCode:
  - LeetCode - Dynamic Programming
- Explore classic dynamic programming problems on GeeksforGeeks:
  - GeeksforGeeks - Dynamic Programming

Practice Questions

1. Solve the 0/1 Knapsack problem.
2. Find the Longest Palindromic Subsequence of a given string.
3. Implement the Matrix Chain Multiplication problem using dynamic programming.
4. Solve the Maximum Subarray Sum problem using dynamic programming.
5. Implement the Longest Common Substring problem using dynamic programming.

Day 5-10

Practice and Advanced Topics

Goal

Strengthen your dynamic programming skills through practice and explore advanced topics.

Topics to cover

- Solve a wide range of dynamic programming problems.
- Explore advanced topics like state compression, multidimensional DP, etc.

Resources

- Solve dynamic programming problems on platforms like LeetCode, HackerRank, and Codeforces.
  - LeetCode - Dynamic Programming
  - HackerRank - Dynamic Programming
  - Codeforces - Dynamic Programming

- Study and research advanced topics in dynamic programming:
    - State Compression: Read about techniques to compress the state space of a dynamic programming problem, such as bitmasking or using efficient data structures.
    - Multidimensional DP: Explore problems that require dynamic programming with multiple dimensions, such as matrix DP or 3D DP.

Practice Questions

1. Solve 3-5 dynamic programming problems each day, focusing on different problem types and techniques.
2. Explore advanced topics and attempt problems related to them, gradually increasing the complexity as you progress.
3. Analyze the time and space complexity of your solutions and optimize them if necessary.