Data Structures and Algorithms (DSA) Practice Introduction

This repository contains a collection of Data Structures and Algorithms (DSA) exercises implemented in Jupyter Notebook. The purpose of this repository is to help you practice and understand fundamental data structures and algorithms concepts, as well as prepare for coding interviews, particularly on platforms like LeetCode.

Table of Contents

Data Structures

Arrays

- Basic array operations (insertion, deletion, searching).
- Array manipulation problems.
- Sliding window techniques.

Linked Lists

- Implementation of singly and doubly linked lists.
- Basic operations (insertion, deletion, traversal).
- Problems related to reversing and merging linked lists.

Stacks and Queues

- Implementation of stacks and queues.
- Problems involving stack and queue operations.
- Evaluating expressions using stacks.

Trees

- Binary Tree and Binary Search Tree (BST) operations.
- Traversals (inorder, preorder, postorder).
- Depth-First Search (DFS) and Breadth-First Search (BFS) algorithms on trees.
- Problems related to tree manipulation.

Hash Tables

- Hashing and collision resolution techniques.
- Implementation of hash tables.
- Solving problems involving hash maps.

Graphs

- Graph representation (adjacency matrix, adjacency list).
- Depth-First Search (DFS) and Breadth-First Search (BFS) on graphs.
- Shortest path algorithms (Dijkstra's, Bellman-Ford).
- Problems related to graph traversal and algorithms.

Algorithms

Sorting

- Popular sorting algorithms (Bubble Sort, Selection Sort, Merge Sort, Quick Sort).
- Analysis of time and space complexity.
- Sorting problems and variations.

Searching

- Linear search and binary search.
- Searching in arrays and other data structures.
- Complex search problems.

Recursion

- Understanding recursion and recursive algorithms.
- Solving problems using recursion.