|  |  |
| --- | --- |
| Design and Analysis of Algorithms | |
| **PART: 01** |  |
|  | [Introduction to Algorithm and Syllabus](https://www.youtube.com/watch?v=u8JZ9gU5o4g&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=1&pp=iAQB) |
|  | [What is Algorithm | How to Analyze an Algorithm | Prior vs Posteriori Analysis](https://www.youtube.com/watch?v=itbkP50iggM&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=2&pp=iAQB) |
|  | [Asymptotic Notations | Big O | Big Omega | Theta Notations](https://www.youtube.com/watch?v=7dz8Iaf_weM&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=3&pp=iAQB) |
|  | [Various Properties of Asymptotic Notation with Examples](https://www.youtube.com/watch?v=OLttwv_4Ltw&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=4&pp=iAQB) |
|  | [Comparison of Various Time Complexities | Different Types in Increasing Order](https://www.youtube.com/watch?v=19N3gWGBh5E&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=5&pp=iAQB) |
|  | [Time Complexities of All Searching and Sorting Algorithm](https://www.youtube.com/watch?v=gBz44smaa9A&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=6&pp=iAQB) |
| - | [Question on Comparison of Various Time Complexities](https://www.youtube.com/watch?v=EH8wQaRI6YE&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=7&pp=iAQB) |
| - | [Question on Comparison of Various Time Complexities](https://www.youtube.com/watch?v=tVJdWy5NKW0&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=8&pp=iAQB) |
|  | [What is Recurrence Relation | How to Write Binary Search Recurrence Relation](https://www.youtube.com/watch?v=NW-naslChdo&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=9&pp=iAQB) |
|  | [Recurrence Relation [T(n)=T(n/2)+c] | Substitution Method](https://www.youtube.com/watch?v=x0n75VFd31U&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=10&pp=iAQB) |
|  | [Recurrence Relation [T(n)=n\*T(n-1)] | Substitution Method](https://www.youtube.com/watch?v=icS-e8RaCyo&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=11&pp=iAQB) |
|  | [Recurrence Relation [T(n)=2T(n/2)+n] | Substitution Method](https://www.youtube.com/watch?v=VHGisohk3Ck&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=12&pp=iAQB) |
|  | [Recurrence Relation [T(n)=T(n-2)+logn] | Substitution Method](https://www.youtube.com/watch?v=JCX8noGdY7k&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=13&pp=iAQB) |
|  | [Recurrence Relation [T(n)=8T(n/2)+n^2] | Master Theorem](https://www.youtube.com/watch?v=FBKjvXGGCJM&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=14&pp=iAQB) |
|  | [Recurrence Relation [T(n)=T(n/2)+c] | Master Theorem](https://www.youtube.com/watch?v=nNabmfua14c&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=15&pp=iAQB) |
|  | [Recurrence Relation [T(n)=T(/n)+logn] | Master Theorem](https://www.youtube.com/watch?v=gazsHlpcfCE&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=16&pp=iAQB) |
|  | [Recurrence Relation [T(n)=2T(n/2)+cn] | Recursive Theorem](https://www.youtube.com/watch?v=bJg_sv7PV-g&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=17&pp=iAQB) |
|  | [Recurrence Relation [T(n)=3T(n/4)+cn^2] | Recursive Theorem](https://www.youtube.com/watch?v=zeVYepdQ9lY&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=18&pp=iAQB) |
|  | [Divide and Conquer Algorithm](https://www.youtube.com/watch?v=I8w2XN0w-fQ&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=19&pp=iAQB) |
|  | [How Quick Sort Words | Performance of Quick Sort with Example](https://www.youtube.com/watch?v=tWCaFVJMUi8&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=20&pp=iAQB) |
|  | [Performance of Quick Sort | Worst Case Time Complexity with Example](https://www.youtube.com/watch?v=eN4RBi4pMvI&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=21&pp=iAQB) |
|  | [How Merge Sort Works, Full Explanation with Example](https://www.youtube.com/watch?v=tn9hxD8gx2M&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=22&pp=iAQB) |
|  | [Merge Sort Pseudocode | Merge Sort with Example](https://www.youtube.com/watch?v=aeI8GGtiLGg&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=23&pp=iAQB) |
|  | [Question on Merge Sort | Divide and Conquer | Algorithm](https://www.youtube.com/watch?v=Ms8g8w-6r2Q&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=24&pp=iAQB) |
|  | [How Bubble Sort Works | Performance of Bubble Sort with Examples](https://www.youtube.com/watch?v=re9ytVtt5zg&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=25&pp=iAQB) |
|  | [Insertion Sort | Time Complexity Analysis | Stable Sort | In-place Sorting](https://www.youtube.com/watch?v=s9fmGjFY1v0&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=26&pp=iAQB) |
|  | [Selection Sort | Time Complexity (Best, Avg and Worst) Analysis](https://www.youtube.com/watch?v=Lrd1QaKyok4&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=27&pp=iAQB) |
|  | [Radix Sort | Easiest Explanation with Example](https://www.youtube.com/watch?v=9QSgBO9yjKU&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=28&pp=iAQB) |
|  | [Counting Sort | Easiest Explanation with Example](https://www.youtube.com/watch?v=mowMVn9wTnE&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=29&pp=iAQB) |
|  | [Bucket Sort | Easiest Explanation | Time Complexity Analysis with Pseudocode](https://www.youtube.com/watch?v=E9OccfF9mpI&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=30&pp=iAQB) |
|  | [Introduction to Trees (Binary Tree, Almost Complete Binary Tree | Full BT | Complete BT)](https://www.youtube.com/watch?v=I_JuQ5ayPmc&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=31&pp=iAQB) |
|  | [Introduction to Heap Tree with Examples | Max Min Heap](https://www.youtube.com/watch?v=uuot9ItgTEI&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=32&pp=iAQB) |
|  | [Insertion in Heap Tree | Max-Heap and Min-Heap Creation | Time Complexity](https://www.youtube.com/watch?v=KzXpfxRzVQM&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=33&pp=iAQB) |
|  | [Question on Max Heap / Min Heap](https://www.youtube.com/watch?v=7gWrUhQxFIU&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=34&pp=iAQB) |
|  | [Build Heap in O(n) Time Complexity | Heapify Method | Full Derivation with Example](https://www.youtube.com/watch?v=8noP3YjjJCM&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=35&pp=iAQB) |
|  | [Deletion in Heap Tree | Time Complexity](https://www.youtube.com/watch?v=4GsxDWMI7tQ&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=36&pp=iAQB) |
|  | [Heap Sort with Example | Heapify Method](https://www.youtube.com/watch?v=nJ6FdAIr_6g&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=37&pp=iAQB) |
|  | [Introduction to Greedy Techniques with Example | What is Greedy Technique](https://www.youtube.com/watch?v=v0eQ4nXJjsk&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=38&pp=iAQB) |
|  | [Knapsack Problem with Example | Greedy Techniques](https://www.youtube.com/watch?v=M79iHjAG1tg&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=39&pp=iAQB) |
|  | [Huffman Coding Algorithm with Example | Greedy Techniques](https://www.youtube.com/watch?v=uDS8AkTAcIU&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=40&pp=iAQB) |
| - | [Question Huffman Coding in Greedy Technique](https://www.youtube.com/watch?v=ocFO1rmh4Zc&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=41&pp=iAQB) |
|  | [Job Sequencing Algorithm with Example | Greedy Techniques](https://www.youtube.com/watch?v=Tpp7o0jQ-8w&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=42&pp=iAQB) |
|  | [Optimal Merge Pattern Using Greedy Method](https://www.youtube.com/watch?v=C7me2iKbty4&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=43&pp=iAQB) |
|  | [What is Spanning Tree with Example](https://www.youtube.com/watch?v=h6cUkkaKNHw&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=44&pp=iAQB) |
|  | [Kruska Algorithm for Minimum Spanning Tree](https://www.youtube.com/watch?v=huQojf2tevI&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=45&pp=iAQB) |
|  | [Prim’s Algorithm for Minimum Cost Spanning Tree](https://www.youtube.com/watch?v=_KX8GDvRzBc&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=46&pp=iAQB) |
|  | [Dijkstra’s Algorithm Single Source Shortest Path – Greedy Method](https://www.youtube.com/watch?v=Gd92jSu_cZk&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=47&pp=iAQB) |
|  | [Dijkstra’s Algorithm Analysis | Time Complexity | Pseudocode Explanation](https://www.youtube.com/watch?v=_0s2e5SqhSk&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=48&pp=iAQB) |
|  | [Why Does Dijkstra Fail on Negative Weights | Full Explanation with Example](https://www.youtube.com/watch?v=R3g2SSlyY_0&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=49&pp=iAQB) |
|  | [Bellman Ford Algorithm | Dijkstra vs Bellman Ford | Single Source Shortest Path](https://www.youtube.com/watch?v=SiI03wnREt4&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=50&pp=iAQB) |
|  | [Bellman Ford Pseudocode and Time Complexity | Single Source Shortest Path](https://www.youtube.com/watch?v=0hXj05XXT7M&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=51&pp=iAQB) |
|  | [BFS and DFS | Breadth First Search | Depth First Search | Graph Traversing | DAA](https://www.youtube.com/watch?v=N2P7w22tN9c&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=52&pp=iAQB) |
|  | [Introduction to Dynamic Programming | Greedy vs Dynamic Programming](https://www.youtube.com/watch?v=0BhhiQGDbEA&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=53&pp=iAQB) |
|  | [0/1 Knapsack Failed Using Greedy Approach](https://www.youtube.com/watch?v=LveF2qwHrqU&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=54&pp=iAQB) |
|  | [0/1 Knapsack Problem | Dynamic Programming | Recursion Equation | Recursion Tree](https://www.youtube.com/watch?v=i8NqAEsZn54&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=55&pp=iAQB) |
|  | [Traveling Salesman Problem | Dynamic Programming](https://www.youtube.com/watch?v=3QiSyc7KyC4&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=56&pp=iAQB) |
|  | [Sum of Subsets Problem | Dynamic Programming](https://www.youtube.com/watch?v=MLSy2_iQdmY&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=57&pp=iAQB) |
|  | [Multistage Graph | Dynamic Program](https://www.youtube.com/watch?v=PE1wsnvbquA&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=58&pp=iAQB) |
|  | [Introduction to All Pair Shortest Path (Floyd Warshall Algorithm)](https://www.youtube.com/watch?v=c5vQS40VBcU&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=59&pp=iAQB) |
|  | [Floyd Warshall Working with Example | All Pair Shortest Path Algorithm](https://www.youtube.com/watch?v=pgNE06YbDZ8&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=60&pp=iAQB) |
|  | [Floyd Warshall Time and Space Complexity | All Pair Shortest Path Algorithm](https://www.youtube.com/watch?v=HAFAD8XbXps&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=61&pp=iAQB) |
|  | [What is Hashing with Example | Hashing in Data Structure](https://www.youtube.com/watch?v=W5q0xgxmRd8&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=62&pp=iAQB) |
|  | [Collision Resolution Techniques in Hashing | What are the Collision Resolution](https://www.youtube.com/watch?v=j612Fj-mgCY&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=63&pp=iAQB) |
|  | [Chaining in Hashing | What is Chaining in Hashing with Example](https://www.youtube.com/watch?v=hmMYPZ5wLX0&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=64&pp=iAQB) |
|  | [Linear Probing in Hashing with Example](https://www.youtube.com/watch?v=ZEyPqqRTO00&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=65&pp=iAQB) |
|  | [Question on Hashing | Linear Probing for Collision in Hash Table](https://www.youtube.com/watch?v=go45eeMrwA4&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=66&pp=iAQB) |
|  | [Quadratic Probing in Hashing with Example](https://www.youtube.com/watch?v=-yQ2Kj-Jn0A&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=67&pp=iAQB) |
|  | [Double Hashing | Collision Resolution Technique](https://www.youtube.com/watch?v=1P7ygNSe9lY&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=68&pp=iAQB) |
|  | [Topological Sorting with Example | Topological Sorting using DFS |](https://www.youtube.com/watch?v=3tkcfvCNtM8&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=69&pp=iAQB) |
|  | [Kahn’s Algorithm | Topological Sorting | DAA](https://www.youtube.com/watch?v=96owfLr89Lk&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=70&pp=iAQB) |
|  | [Branch and Bound Algorithm with Example | Easiest Explanation of B&B with Example](https://www.youtube.com/watch?v=XZbrmetb9VE&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=71&pp=iAQB) |
|  | [0/1 Knapsack Using Branch and Bound with Example](https://www.youtube.com/watch?v=CwM-Mv0Bm4Y&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=72&pp=iAQB) |
|  | [Shortcut for Swaps in Bubble Sorting | Best Case | Worst Case](https://www.youtube.com/watch?v=2jhN6aazfSQ&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=73&pp=iAQB) |
|  | [Optimized Bubble Sort | Best Case | O(n) Time Complexity | Shorting Algorithm](https://www.youtube.com/watch?v=g6hr8B3OWio&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=74&pp=iAQB) |
|  | [Recurrence Relation [T(n)=2T(n/2)+2] Min-Max Algorithm](https://www.youtube.com/watch?v=EnIuY1GpG3g&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=75&pp=iAQB) |
|  | [Min-Max Algorithm without Divide and Conquer | Linear Approach | Find Max and Min](https://www.youtube.com/watch?v=rJAsiBW0oZ0&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=76&pp=iAQB) |
|  | [Min-Max Algorithm with Divide and Conquer](https://www.youtube.com/watch?v=g8DJiGWoQmk&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=77&pp=iAQB) |
|  | [Time Complexity and Comparison of All Data Structures](https://www.youtube.com/watch?v=1lqoJ-NSmDE&list=PLxCzCOWd7aiHcmS4i14bI0VrMbZTUvlTa&index=78&pp=iAQB) |

|  |  |
| --- | --- |
| Data Structure | |
| **PART: 01** |  |
|  | [Roadmap of DSA Syllabus](https://www.youtube.com/watch?v=qNGyI95E5AE&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=1&pp=iAQB) |
|  | [Introduction to Data Structure with Real Life Examples](https://www.youtube.com/watch?v=3cU__spdMIw&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=2&pp=iAQB) |
|  | [Arrays in Data Structure | Initialization, Declaration, Memory Representation](https://www.youtube.com/watch?v=6e6yKtr2VGI&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=3&pp=iAQB) |
|  | [Types of Array | One Dimensional and Multi-dimensional Array](https://www.youtube.com/watch?v=lAEmhJA-tVw&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=4&pp=iAQB) |
|  | [Addressing in One Dimensional Array](https://www.youtube.com/watch?v=JTfPmCiLhz0&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=5&pp=iAQB) |
|  | [Arrays | Addressing in 2D Arrays | Row Major Order](https://www.youtube.com/watch?v=BNA6Sb1wM8E&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=6&pp=iAQB) |
|  | [3D Arrays | Addressing in 3D Arrays | Row Major Order](https://www.youtube.com/watch?v=mzfhcfa0Ra0&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=7&pp=iAQB) |
|  | [Addressing in Lower Triangular Matrix](https://www.youtube.com/watch?v=X-vk__Y1Tgk&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=8&pp=iAQB) |
|  | [Find 2nd Largest Number in Array](https://www.youtube.com/watch?v=eIm9ODShmCI&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=9&pp=iAQB) |
|  | [Two Pointer Technique | Two Sum Problem in Data Structure](https://www.youtube.com/watch?v=B2L4mAglJZA&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=10&pp=iAQB) |
|  | [Maximum Sum Subarray Problem | Understand Naïve Approach](https://www.youtube.com/watch?v=9N3hDuS28EI&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=11&pp=iAQB) |
|  | [Sliding Window Technique](https://www.youtube.com/watch?v=uqGxFk0cEdI&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=12&pp=iAQB) |
|  | [Remove Duplicate Elements in Sorted Array | Various Methods](https://www.youtube.com/watch?v=f1BTfBv22og&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=13&pp=iAQB) |
|  | [Linear Search in Data Structure | Time Complexity](https://www.youtube.com/watch?v=t9gErvSVid0&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=14&pp=iAQB) |
|  | [Binary Search in Data Structure](https://www.youtube.com/watch?v=9HlbVEVt_Y0&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=15&pp=iAQB) |
|  | [Introduction to Linked List | Types and Need of Linked List](https://www.youtube.com/watch?v=Y3Ckd3OW0_g&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=16&pp=iAQB) |
|  | [Arrays vs Linked List](https://www.youtube.com/watch?v=d4pp0noHOB8&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=17&pp=iAQB) |
|  | [Single Linked List in Data Structure | Self Referential Structure](https://www.youtube.com/watch?v=M1SlAsRCZvM&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=18&pp=iAQB) |
|  | [The Basic Operations of Linked List](https://www.youtube.com/watch?v=0XKlbNJLvjs&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=19&pp=iAQB) |
|  | [Traversing in Linked List](https://www.youtube.com/watch?v=I16sYpAwBEg&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=20&pp=iAQB) |
|  | [Insert a Node At The Beginning of Linked List](https://www.youtube.com/watch?v=v5QzD4Rkar8&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=21&pp=iAQB) |
|  | [Insert a Node At the End of Linked List](https://www.youtube.com/watch?v=ZTTy6jTTJUI&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=22&pp=iAQB) |
|  | [Insert a Node After a Given Node in Linked](https://www.youtube.com/watch?v=B6_uo7KA20M&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=23&pp=iAQB) |
|  | [Delete a Node From The Beginning of Linked List](https://www.youtube.com/watch?v=bGU37FZnN7c&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=24&pp=iAQB) |
|  | [Delete a Node From The End of Linked List](https://www.youtube.com/watch?v=dzmnI9hCk6E&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=25&pp=iAQB) |
|  | [Deletion After a Given Node in Linked List](https://www.youtube.com/watch?v=n8kLcrzsuec&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=26&pp=iAQB) |
|  | [Deleting Entire Linked List | Single Linked List](https://www.youtube.com/watch?v=-AXzieE0umw&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=27&pp=iAQB) |
|  | [Introduction to Double Linked List](https://www.youtube.com/watch?v=818gGdKTB_U&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=28&pp=iAQB) |
|  | [How to Find a Cycle in Linked List | 2 Pointer Algorithm](https://www.youtube.com/watch?v=f-JWjs8NtFQ&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=29&pp=iAQB) |
|  | [Introduction to Stack | PUSH and POP Operations](https://www.youtube.com/watch?v=HRDHgKrYHgU&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=30&pp=iAQB) |
|  | [PUSH Operation Using Array | Implementation of Stack](https://www.youtube.com/watch?v=rR0WheIxDC0&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=31&pp=iAQB) |
|  | [POP Operation Using Array | Implementation of Stack](https://www.youtube.com/watch?v=k8gKrc_aaQc&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=32&pp=iAQB) |
|  | [PUSH Operation Using Linked List | Implementation of Stack](https://www.youtube.com/watch?v=S0Z7K-5HG_Y&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=33&pp=iAQB) |
|  | [POP Operation Using Linked List | Implementation of Stack](https://www.youtube.com/watch?v=Ava0zesUeik&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=34&pp=iAQB) |
|  | [Recursion vs Loop | How Both Approaches Work](https://www.youtube.com/watch?v=J4ucdS7uvls&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=35&pp=iAQB) |
|  | [Infix, Prefix and Postfix with Examples](https://www.youtube.com/watch?v=vtyuftvugXY&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=36&pp=iAQB) |
|  | [Infix to Prefix Conversion | Infix to Prefix Conversion with Examples](https://www.youtube.com/watch?v=f6xHar7x_SQ&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=37&pp=iAQB) |
|  | [Infix to Postfix Conversion | Infix to Postfix Conversion with Examples](https://www.youtube.com/watch?v=41FCbf7IbUY&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=38&pp=iAQB) |
|  | [Infix to Prefix Conversion Using Stack](https://www.youtube.com/watch?v=E0QGRN-zi_c&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=39&pp=iAQB) |
|  | [Infix to Postfix Conversion Using Stack](https://www.youtube.com/watch?v=N1Sr-j-cf4c&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=40&pp=iAQB) |
|  | [Question on Infix to Postfix notation](https://www.youtube.com/watch?v=aOa9Jo2Msnk&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=41&pp=iAQB) |
|  | [Postfix Expression Evaluation | Stack Application](https://www.youtube.com/watch?v=dc4bfoxIbu8&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=42&pp=iAQB) |
|  | [Question On Postfix Evaluation | Stack](https://www.youtube.com/watch?v=a9IXpAy827c&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=43&pp=iAQB) |
|  | [Understanding Call Stack with Example](https://www.youtube.com/watch?v=x9yuHFyuWZg&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=44&pp=iAQB) |
|  | [The Magic Recursion: Understanding The Power of Recursive Function](https://www.youtube.com/watch?v=E3i5Tm9k38s&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=45&pp=iAQB) |
|  | [Introduction to Queue Data Structure with Real Life Example](https://www.youtube.com/watch?v=YMzZTTO2MpE&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=46&pp=iAQB) |
|  | [Enqueue(), Dequeue() and Other Operations on Queue](https://www.youtube.com/watch?v=jvSdGoNnQo4&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=47&pp=iAQB) |
|  | [Dequeue() in Queue Using Array](https://www.youtube.com/watch?v=PpfHfB8CSWc&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=48&pp=iAQB) |
|  | [Isfull() and isempty() in Queue | Queue Operations](https://www.youtube.com/watch?v=2K1qUZy-hAw&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=49&pp=iAQB) |
|  | [Implementation of Queue Using Array | Enqueue() in Queue](https://www.youtube.com/watch?v=hGbphQIOHYs&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=50&pp=iAQB) |
|  | [Implementation of Queue Using Linked List | Euqueue() in Queue](https://www.youtube.com/watch?v=XqtbBB-Y-Zs&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=51&pp=iAQB) |
|  | [Introduction to Trees | Binary Tree, Almost Complete Binary Tree | Full BT](https://www.youtube.com/watch?v=z0Vnno96_MA&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=52&pp=iAQB) |
|  | [Binary Search Tree in Data Structure | Insertion and Traversal in BST](https://www.youtube.com/watch?v=sXABdGalFNg&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=53&pp=iAQB) |
|  | [Deletion From Binary Search Tree (BST) with Example](https://www.youtube.com/watch?v=hL9RUD33nYs&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=54&pp=iAQB) |
|  | [Find Preorder, Post-Order, In-Order of Binary Search Tree (BST)](https://www.youtube.com/watch?v=uI77Ij5Kiic&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=55&pp=iAQB) |
|  | [Preorder, In-Order and Post-Order in 5 Minute | Tree Traversal | Easiest and Shortest](https://www.youtube.com/watch?v=XRcC7bAtL3c&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=56&pp=iAQB) |
|  | [Question on Binary Search Tree](https://www.youtube.com/watch?v=XU2YE1CLDMk&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=57&pp=iAQB) |
|  | [Introduction to AVL Tree in Data Structure with Example](https://www.youtube.com/watch?v=E9DOBLNB-aE&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=58&pp=iAQB) |
|  | [How to Crate AVL Tree | LL, RR, LR, RL Rotation in AVL](https://www.youtube.com/watch?v=vEZ3-bdPdUQ&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=59&pp=iAQB) |
|  | [AVL Tree Creation in Data Structure | All Points](https://www.youtube.com/watch?v=VKV8PYhcQoo&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=60&pp=iAQB) |
|  | [Time Complexities of All Trees | Binary Tree, Binary Search Tree, AVL Tree, Heap Tree](https://www.youtube.com/watch?v=-qqGce9Cijw&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=61&pp=iAQB) |
|  | [Introduction to Red-Black Tree](https://www.youtube.com/watch?v=hq_3iqS0Hq0&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=62&pp=iAQB) |
|  | [Insertion in Red Black Tree](https://www.youtube.com/watch?v=tvY0Xv5dVuc&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=63&pp=iAQB) |
|  | [Topological Order / Sort in DAG (Direct Acyclic Graph)](https://www.youtube.com/watch?v=ChxA6H2253I&list=PLxCzCOWd7aiEwaANNt3OqJPVIxwp2ebiT&index=64&pp=iAQB) |

|  |  |
| --- | --- |
| Design Patterns | |
| **PART: 01** |  |
|  | [What are Design Patterns | Introduction to Design Patterns and Principles](https://www.youtube.com/watch?v=mE3qTp1TEbg&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=1&pp=iAQB) |
|  | [The Singleton Pattern Explained and Implemented in Java | Creational Pattern](https://www.youtube.com/watch?v=tSZn4wkBIu8&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=2&pp=iAQB) |
|  | [The Factory Method Pattern Explained and Implemented in Java | Creational Pattern](https://www.youtube.com/watch?v=EdFq_JIThqM&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=3&pp=iAQB) |
|  | [The Abstract Factory Patten Explained and Implemented | Creational Pattern](https://www.youtube.com/watch?v=QNpwWkdFvgQ&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=4&pp=iAQB) |
|  | [The Builder Pattern Explained and Implemented in Java | Creational Pattern](https://www.youtube.com/watch?v=MaY_MDdWkQw&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=5&pp=iAQB) |
|  | [The Prototype Pattern Explained and Implemented in Java | Creational Pattern](https://www.youtube.com/watch?v=DcFhITC9v0E&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=6&pp=iAQB) |
|  | [The Chain of Responsibility Pattern Explained and Implemented | Behavioral Pattern](https://www.youtube.com/watch?v=FafNcoBvVQo&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=7&pp=iAQB) |
|  | [The Command Pattern Explained and Implemented in Java | Behavioral Pattern](https://www.youtube.com/watch?v=UfGD60BYzPM&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=8&pp=iAQB) |
|  | [The Template Method Pattern Explained Implemented in Java | Behavioral Pattern](https://www.youtube.com/watch?v=cGoVDzHvD4A&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=9&pp=iAQB) |
|  | [The Mediator Pattern Explained and Implemented in Java | Behavioral Pattern](https://www.youtube.com/watch?v=35D5cBosD4c&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=10&pp=iAQB) |
|  | [The Memento Pattern Explained and Implemented in Java | Behavioral Pattern](https://www.youtube.com/watch?v=_Q5rXfGuyLQ&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=11&pp=iAQB) |
|  | [The Observer Pattern Explained and Implemented in Java | Behavioral Pattern](https://www.youtube.com/watch?v=-oLDJ2dbadA&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=12&pp=iAQB) |
|  | [The State Pattern Explained and Implemented in Java | Behavioral Pattern](https://www.youtube.com/watch?v=abX4xzaAsoc&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=13&pp=iAQB) |
|  | [The Iterator Pattern Explained and Implemented in Java | Behavioral Pattern](https://www.youtube.com/watch?v=QCWJWfuAfJc&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=15&pp=iAQB) |
|  | [The Visitor Pattern Explained and Implemented in Java | Behavioral Pattern](https://www.youtube.com/watch?v=UQP5XqMqtqQ&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=16&pp=iAQB) |
|  | [The Adapter Pattern Explained and Implemented in Java | Structural Pattern](https://www.youtube.com/watch?v=wA3keqCeKtM&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=17&pp=iAQB) |
|  | [The Bridge Pattern Explained and Implemented in Java | Structural Pattern](https://www.youtube.com/watch?v=88kAIisOiYs&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=18&pp=iAQB) |
|  | [The Composite Pattern Explained and Implemented in Java | Structural Pattern](https://www.youtube.com/watch?v=oo9AsGqnisk&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=19&pp=iAQB) |
|  | [The Decorator Pattern Explained and Implemented in Java | Structural Pattern](https://www.youtube.com/watch?v=v6tpISNjHf8&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=20&pp=iAQB) |
|  | [The Façade Pattern Explained and Implemented in Java | Structural Pattern](https://www.youtube.com/watch?v=xWk6jvqyhAQ&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=21&pp=iAQB) |
|  | [The Flyweight Pattern Explained and Implemented in Java | Structural Pattern](https://www.youtube.com/watch?v=qscOsQV-K14&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=22&pp=iAQB) |
|  | [The Proxy Pattern Explained and Implemented in Java | Structural Pattern](https://www.youtube.com/watch?v=TS5i-uPXLs8&list=PLlsmxlJgn1HJpa28yHzkBmUY-Ty71ZUGc&index=23&pp=iAQB) |