Section Overview, What Is A Data Structure?, How Computers Store Data, Data Structures In Different Languages, Operations On Data Structures,

Arrays Introduction, Static vs Dynamic Arrays, Quick Note: Upcoming Video, Optional: Classes In Javascript, Implementing An Array, Strings and Arrays, Exercise: Reverse A String, Solution: Reverse A String,

Hash Tables Introduction, Hash Function, Hash Collisions, Hash Tables In Different Languages, Exercise: Implement A Hash Table, Solution: Implement A Hash Table, keys(), Extra: keys() Without Collision, Hash Tables VS Arrays, HashMap in detail.

Linked Lists Introduction, What Is A Linked List?, Exercise: Imposter Syndrome, Exercise: Why Linked Lists?, Solution: Why Linked Lists?, What Is A Pointer?, Our First Linked List, Solution: append(), Solution: prepend(), Node Class, insert(), Quick Note: Upcoming Video, Solution: insert(), Solution: remove(), Doubly Linked Lists, Exercise: Doubly Linked Lists, Solution: Doubly Linked Lists, Singly VS Doubly Linked Lists,

left: append(), Solution: prepend(),

Exercise: reverse(), Solution: reverse(), Linked Lists Review, Stacks + Queues Introduction, Stacks, Queues, Exercise: Stacks VS Queues, Solution: Stacks VS Queues, Quick Note: Upcoming Video, Optional: How Javascript Works, Exercise: Stack Implementation (Linked Lists), Solution: Stack Implementation (Linked Lists), Exercise: Stack Implementation (Array), Solution: Stack Implementation (Array), Exercise: Queue Implementation,

Trees Introduction, Binary Trees, O(log n), Binary Search Trees, Balanced VS Unbalanced BST, BST Pros and Cons, Exercise: Binary Search Tree, Solution: insert(), Solution: lookup(), Extra Exercise: remove(), Solution: remove(), AVL Trees + Red Black Trees, Resources: AVL Trees + Red Black Trees, Priority Queue,

Graphs Introduction, Types Of Graphs, Exercise: Guess The Graph, Graph Data, Exercise: Graph Implementation, Solution: Graph Implementation, Graphs Review, Data Structures Review, What Else Is Coming Up?,

Introduction to Algorithms, Recursion Introduction, Stack Overflow, Anatomy Of Recursion, Exercise: Factorial, Solution: Factorial, Exercise: Fibonacci, Solution: Fibonacci, Recursive VS Iterative, When To Use Recursion, Exercise: Reverse String With Recursion,

left:, Recursion Introduction,Fibonacci,Factorial

Sorting Introduction, The Issue With sort(), Sorting Algorithms, Bubble Sort, Exercise: Bubble Sort, Solution: Bubble Sort, Selection Sort, Exercise: Selection Sort, Solution: Selection Sort, Dancing Algorithms, Insertion Sort, Exercise: Insertion Sort, Solution: Insertion Sort, Merge Sort and O(n log n), Exercise: Merge Sort, Solution: Merge Sort, Stable VS Unstable Algorithms, Quick Sort, Optional Exercise: Quick Sort, Which Sort Is Best?, Resources: Heap Sort, Radix Sort + Counting Sort, Resources: Radix Sort + Counting Sort, Exercise: Sorting Interview, Solution: Sorting Interview, Sorting In Your Language

left: O(n log n),

Searching + Traversal Introduction, Linear Search, Binary Search, Graph + Tree Traversals, BFS Introduction, DFS Introduction, BFS vs DFS, Resources: BFS vs DFS, Exercise: BFS vs DFS, Solution: BFS vs DFS, breadthFirstSearch(), breadthFirstSearchRecursive(), PreOrder, InOrder, PostOrder, depthFirstSearch(), Optional Exercise: Validate A BST, Graph Traversals, BFS in Graphs, DFS in Graphs, Dijkstra + Bellman-Ford Algorithms, Searching + Traversal Review, Dynamic Programming Introduction, Memoization 1, Memoization 2

1 1 2 1 3 4 6 =2

Fibonacci and Dynamic Programming, Dynamic Programming, Implementing Dynamic Programming