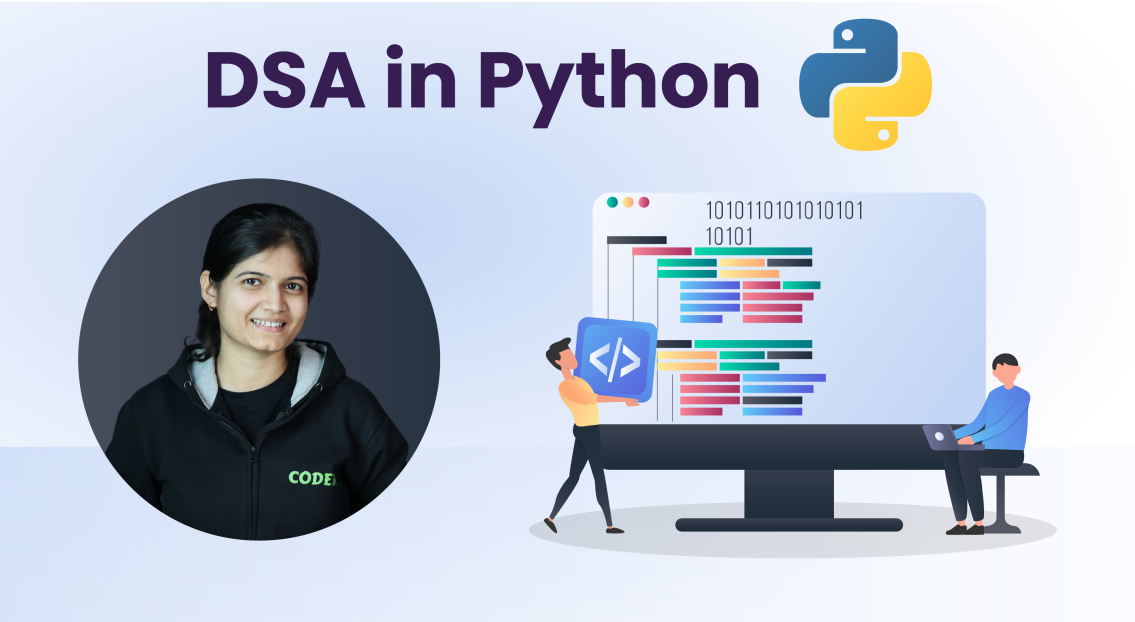
Welcome to ineuron.ai



Complete DSA in Python

Description:

A comprehensive chase to excel any intreview for the Data Structures and Algorithms. This course has been specifically designed to provide resources that would assist you in cracking problem-solving interviews. The presented problems in the course would suffice to look on to positive outcomes in the interviews.

Start Date:

Doubt Clear Time:

Course Time: Flexible

Features:

Course Materials

Self Paced Learning

Lifetime Dashboard Access

Completion Certificate

What we learn:

Introduction to Algorithms

Analysis in Algorithms

Array Data Structure

Heap Data Structure

Recursion

Divide and Conquer

Linked List Data Structure

Stack and Queue

Hashing Data Structure

Tree Data Structure

Binary Search Tree

Graph Traversal Algorithms

Application of greedy algorithm

Dynamic Programming

Research Area- P, NP, NP-Hard and NP-Complete Problems

Requirements:

System with minimum i3 processor or better

At least 4 GB of RAM

Working internet connection

Dedication to learn

Instructor:

Name:

Priya Bhatia

Description:

Expertise in data structure competitive programing and solving an analytical problems and implementing data structure algorithm in multiple programing language. I have done my M.Tech in Artificial Intelligence at IIT Hyderabad and have an experience of implementation in multiple projects.

>Introduction to Algorithms:

>>Complete DSA Roadmap

>>Why DSA required

>>Algorithms Introduction

>>Steps to construct an algo

>Analysis in Algorithms:

>>Types of Analysis

>>Asymptotic Notation - Big O Time Complexity

>>Asymptotic Notation - Omega Time Complexity

>>Asymptotic Notation - Theta Time Complexity

>>Apriori Analysis - Time Complexity Analysis Part1

>>Apriori Analysis - Time Complexity Analysis Part2

>>Apriori Analysis - Time Complexity Analysis Part3

>>Practice Set - Asymptotic Notations

>>Complexity Classes

>>Recurrence Relation Introduction

>>Substitution Method - Problem 1

>>Substitution Method - Problem 2

>>Substitution Method - Problem 3

>>Recursive Tree Approach - Problem 1

>>Recursive Tree Approach - Problem 2

>>Recursive Tree Approach - Problem 3

>>Practice Set - Substitution and Recursive Tree Approach

>>Masters Theorem Case 1

>>Masters Theorem Case 2

>>Masters Theorem Case 3

>>Practice Set - Masters Theorem

>Array Data Structure:

>>Introduction to Array Data Structure

>>Array Data Structure Implementation

>>Address of an element in 1D array

>>Address of an element in 2D array

>>Searching of an element - Linear Search

>>Searching of an element - Binary Search

>>Recurrence Relation of Binary Search

>>Implementation of Binary Search

>>Binary Search Interview Problem

>>Search a 2D Matrix

>>Searching of an element - Ternary Search

>>Recurrence Relation of Ternary Search

>>Implementation of Ternary Search

>>Sorting in an array - Comparison and Non-Comparison

>>Stable and Unstable sorting algorithms

>>Inplace and Outplace Sorting algorithms

>>Comparison Sort - Bubble Sort

>>Comparison Sort - Bubble Sort Implementation

>>Comparison Sort - Selection Sort

>>Comparison Sort - Selection Sort Implementation

>>Comparison Sort - Insertion Sort

>>Comparison Sort - Insertion Sort Implementation

>>FAANG Interview Question on Arrays - Best Time to Buy and Sell Stock

>>FAANG Interview Question on Arrays - Collinear Points

>>FAANG Interview Question on Arrays - Majority Element

>>FAANG Interview Question on Arrays - Sort Colors

>Heap Data Structure:

>>Basics of Heap Sort - Full Binary Tree vs Complete Binary Tree vs Almost Complete

>>Concept of Minheap and Maxheap Tree

>>Insertion in Minheap or Maxheap Tree

>>Deletion in Minheap or Maxheap Tree

>>Creation of Minheap or Maxheap Tree

>>Time Complexity Derivation to build minheap or maxheap

>>Comparison Sort - Heap Sort

>>FAANG Interview Question on Heap - Top K frequent elements

>>FAANG Interview Question on Heap - K Closest Points to Origin

>Recursion:

>>Introduction to Recursion

>>Factorial Finding using Recursion with its Implementation

>>Fibonacci Series using Recursion with its Implementation

>>Count Of number of ways to reach upstairs

>Divide and Conquer:

>>Introduction to Divide and Conquer

>>Applications of Divide and Conquer - Finding of maxima and minima

>>Applications of Divide and Conquer - Implementation of finding of maxima and minim

>>Applications of Divide and Conquer - Finding of power of an element with its Impleme

>>Applications of Divide and Conquer - Binary Search

>>Applications of Divide and Conquer - Recurrence relation of Binary Search

>>Applications of Divide and Conquer - Implementation of Binary Search

>>FAANG Interview Question- Two Pointers Problem

>>Applications of Divide and Conquer - Merge Sort

>>Applications of Divide and Conquer - Implementation of Merge Sort

>>FAANG Interview Question on MergeSort - Finding of single sorted array complexity

>>Applications of Divide and Conquer - Quick Sort

>>Applications of Divide and Conquer - Implementation of Quick Sort

>>FAANG Interview Scenario Based Question on QuickSort complexity

>>Applications of Divide and Conquer - Randomized QuickSort

>>Applications of Divide and Conquer - Selection Procedure

>>Applications of Divide and Conquer - Implementation of Selection Procedure

>>Applications of Divide and Conquer - Count Of number of an inversions

>>Applications of Divide and Conquer - Strassen's Matrix Multiplication

>Linked List Data Structure:

>>Introduction to Linked List

>>Insertion of a node in Linked List - Front

>>Insertion of a node in Linked List - After a given node

>>Insertion of a node in Linked List - End

>>Deletion of a node in Linked List

>>Searching of a node in Linked List

>>FAANG Interview Question - Reversal of a node in Linked List

>>FAANG Interview Question - Count of all nodes in Linked List

>>FAANG Interview Question - Floyd's Cycle Detection Algorithm

>>FAANG Interview Question - Merge Of two Sorted Linked List

>Skip List Data Structure:

>>Skip List- Motivation, Build-in, Search, Insertion and Deletion skip list

>Stack and Queue:

>>Introduction to Stack Data Structure and Push Operation in depth

>>Stack- Pop operation

>>Implementation of Stack using array and linked list

>>Queue- Insertion and Deletion operation

>>Implementation of Queue using array and linked list

>>FAANG Interview Question - Valid Parenthesis

>Hashing Data Structure:

>>Introduction to Hashing Data Structure

>>Hash Function and its types

>>Implementation of Hash Functions

>>Open addressing - Linear Probing and Primary Clustering

>>Open addressing - Quadratic Probing and Secondary Clustering

>>Open addressing - Double Hashing

>>Chaining

>>Load Factor and Rehashing

>Tree Data Structure:

>>Basics of Tree - Full Binary Tree vs Complete Binary Tree vs Almost Complete Binary

>Tree Traversal Algorithms:

>>Tree Traversal Algorithms- Inorder, Preorder and PostOrder

>>FAANG Interview Questions on Tree Traversal Algorithm

>Binary Search Tree:

>>Introduction to Binary Search Tree

>>Insertion and Inorder Traversal in BST

>>FAANG Interview Question- Minimum value in BST

>>FAANG Interview Question- Find unique possible BST's

>>Searching in Binary Search Tree

>>Deletion in Binary Search Tree

>Graph Traversal Algorithms:

>>Basics Of Graph- Simple vs Multigraph, Null vs Complete Graph, Relationship betwe

>>Introduction to Graph Traversal Algorithms

>>Introduction to Depth First Search

>>DFS Psuedocode and illustration using an example

>>DFS Coding Implementation

>>BFS Intro, Psuedocode and illustration using an example

>>BFS Coding Implementation

>Greedy Algorithm:

>>Introduction to greedy algorithm

>Application of greedy algorithm:

>>Fractional Knapsack Problem

>>Implementation of Fractional Knapsack Problem

>>Basics Of Graph- Simple vs Multigraph, Null vs Complete Graph, Relationship betwe

>>Introduction to Spanning Tree and Minimum Spanning Tree

>>Minimum Spanning Tree- Kruskal 's Algorithm

>>Minimum Spanning Tree- Prim's Algorithm

>>Single Source Shortest Path- Dijkstra's algorithm

>>Single Source Shortest Path- Dijkstra's algorithm Implementation

>>Huffman Coding

>>Optimal Merge Pattern

>>Job Sequencing with Deadline

>Dynamic Programming:

>>Introduction to Dynamic Programming

>Application of Dynamic Programming:

>>Fibonacci Series using Dynamic Programming

>>0-1 Knapsack Problem

>Research Area- P, NP, NP-Hard and NP-Complete Problems:

>>Research Area- P, NP, NP-Hard and NP-Complete Problems

>Some ending tips for all students:

>>Some ending tips for all students

>Detailed Interview Process to

crack FAANG Companies(SDE Roles):

>>Detailed Interview Process to crack FAANG Companies