***Python***

***Module 1: Basics-I***

1. Python Introduction.
2. Python History.
3. Python applications.
4. Python IDE
5. Python Text editors
6. Python Installation
7. Keywords
8. Variables
9. Identifiers
10. Literals
11. Comments and Docstring
12. What is Input
13. Data Types Introduction
14. Int Datatype
15. Int Methods
16. String Datatype
17. String Index & Slicing Using Positive & Negative Index.
18. String Methods
19. Complex datatype
20. Complex methods
21. Float datatype
22. Float methods
23. Boolean Datatype
24. List Datatype
25. List Methods
26. Slice a List Using Positive & Negative Index
27. How To Take Input for List
28. List Advantages
29. Tuple Datatype
30. Tuple Methods
31. Tuple a List Using Positive & Negative Index
32. How to take input for tuples
33. Why tuples are faster than list
34. Set datatype
35. Set Methods
36. How to take Input for Set
37. Set Advantages and Disadvantages
38. Dictionary data type
39. Dictionary Methods
40. Frozenset Datattype
41. Frozenset methods

***Module 2: Basics-II***

1. Type casting & Type conversion
2. Flow control statements
3. Conditional control statements
4. Simple if statement
5. if else statement
6. if elif else statement
7. nested if statement
8. Repetitive statements
9. For loop
10. While loop
11. Nested For loop
12. Loop Control Statements
13. break statement
14. continue statement
15. pass statement
16. for else & while else statement
17. Operatorsin python
18. Arithmetic operators
19. Assignment operators
20. Comparison operators
21. Logical or Boolean Operators
22. Bitwise Operators
23. Membership operators
24. Identity Operators
25. Operator precedence & Associativity
26. Operators used for Strings
27. Operators used for List
28. Operators used for Tuples
29. Operators used for Set
30. Operators used for Dictionary

***Module 1: Basics-III***

1. Functions Introduction
2. Types of arguments in functions
3. Scope of a variable
4. LEGB rule
5. Global vs Nonlocal
6. Built in Functions Introduction
7. Lambda Function
8. Print Function
9. Map Function
10. Filter Function
11. What are Modules
12. Types of Modules
13. Built in modules
14. math module
15. string module
16. random module
17. Array introduction
18. Array indexing
19. Array Slicing
20. Methods on array
21. Built in functions on arrays
22. operators used for arrays
23. file handling
24. comprehension techniques
25. Exception handling

***Module 3: Intermediate***

1. Object oriented programming
2. Why we need oops
3. OOPs concepts
4. Class & Object
5. Inheritance
6. Operator overloading
7. Method overloading
8. Method overriding
9. Polymorphism
10. Encapsulation
11. Name Mangling
12. Data Hiding
13. Abstraction
14. Static variable vs Instance variable
15. Instance method& Class Method& Static method
16. Dunder methods
17. How to create Dunder methods

***Module 4: Python Advanced***

1. Threading
2. Magic Methods
3. Generators
4. Decorators
5. Regular Expressions
6. Tkinter

***Data Structure and Algorithms***

#### 1. Introduction session- 1

##### **1.1 - Introduction to array and list**

##### **1.2 – Array Problem Find pair whose sum equal to given target Approach Discussion**

##### **1.3 – Binary Search**

##### **1.4 – Rotate an Array in clockwise**

##### **1.5 – Search in a sorted Rotated array**

#### 2. Arrays

##### **2.1 – Introduction to Arrays and Lists in Python 1**

##### **2.2 – List Methods and Functions in Python 1**

##### **2.3 – Sorting element in a List Bubble Sort 1**

##### **2.4 – Binary Search Explanation**

#### 3. Complexity Discussion + Example

##### **3.1 – Introduction to Time Complexity**

##### **3.2 – Examples**

##### **3.3 – Big O Notation**

##### **3.4 – Other Asymptotic Notations**

##### **3.5 – Examples and Space Complexity**

#### 4. Complexity + Binary Search + Recursion

##### **4.1 – Doubt Session**

##### **4.2 – Binary Search**

##### **4.3 – Introduction to Recursion**

#### 5. Complexity + Quick Short

##### **5.1 – Doubts Discussion of Assignment**

##### **5.2 – Recursion**

##### **5.3 – Binary Search using Recursion**

##### **5.4 – Quick Sort Algorithm**

#### 6. Merge Short, generate sum from 2 number Maximum sum of k sized sub Array, Sliding window

##### **6.1 – 2-Sum Problem**

##### **6.2 – Print all pairs that sum to target**

##### **6.3 – Maximum sum for K sized subarray (Sliding Window)**

##### **6.4 – Maximum sum of a given subarray (Kadane’s Algorithm)**

#### 7. Linked List

##### **7.1 – Time Complexity of Quick Sort and Merge Sort**

##### **7.2 – Introduction to Linked List.mp4**

##### **7.3 – Insertion in Linked List at the end**

##### **7.4 – Delete Last Element in Linked List**

##### **7.5 – Insert and Delete at any Position**

##### **7.6 – Cycle Detection in Linked List**

#### 8. Linked List Loop Detection, Doubly Liked List

##### **8.1 – Cycle Detection in Linked List**

##### **8.2 – Circular and Doubly Linked List**

##### **8.3 – Reverse a Singly Linked List**

##### **8.4 – Applications of Reversing a Linked List**

#### 9. Stack’s Next Greater Element, Balanced Brackets, Previous Greater Element, List with Random

##### **9.1 – Introduction to Stacks**

##### **9.2 – Next Greater Element on Right**

##### **9.3 – Previous Greater Element on Left**

##### **9.4 – Balanced Brackets Problem**

##### **9.5 – Doubts Discussion**

#### 10. Trapping Rain Water, Min stack, Queue, Stack to Queue, Queue to Stack

##### **10.1 – Trapping Rain Water Problem**

##### **10.2 – Minimum Stack**

##### **10.3 – Introduction To Queues**

##### **10.4 – Stack To Queue**

##### **10.5 – Queue To Stack**

#### 11. Stock Snap Problem, Histogram Problem

##### **11.1 – Stock Span Problem**

##### **11.2 – Max Area Histogram and Storage of Rain**

#### 12. Priority Queue

##### **12.1 – Queue and Priority Queue**

##### **12.2 – Introduction to Trees**

##### **12.3 – Tree Traversals**

#### 13. Binary Tree, Pre Order Traversal, Post Order, In Order, Height of Tree, Insertion in Tree

##### **13.1 Tree Traversal BFS (Level Order)**

##### **13.2 Tree Traversal DFS (Preorder, In order, Post Order)**

##### **13.3 Height of Tree**

##### **13.4 Insertion and Deletion in BST**

#### 14. LCA Binary Tree, print all root to leaf Path, Diameter of Tree

##### **14.1 – LCA in Binary Tree**

##### **14.2 – Diameter of Tree**

##### **14.3 – Maximum path sum**

##### **14.4 – Introduction To Tries**

#### 15. Sum Tree, Left view, right view

##### **15.1 – Sum Tree**

##### **15.2 – Number of Paths from root to leaf**

##### **15.3 – Left View of Binary Tree**

##### **15.4 – Right View of Binary Tree**

##### **HON 15.5 – Top View and Bottom View of Binary Tree**

##### **15.6 – LCA of BST**

#### 16. Trie, Heaps

##### **16.1 Tries**

##### **16.2 Heaps**

#### 17. Heapify, heapsort, Back Tracking

##### **17.1 Heaps Implementation**

##### **17.2 Heap Sort and Heapify Time Complexity**

##### **17.3 Introduction to Backtracking**

#### 18. Maze Solver, N Queen Problem

##### **18.1 – Maze solver discussion.mp4**

##### **18.2 – Maze solver Coding**

##### **18.3 – N-Queen**

##### **18.4 – Sudoku Solver**

#### 19. Greedy Algorithm, Activiti Selection, Huffman encoding

##### **19.1 – Greedy Algorithms Discussion**

##### **19.2 – Activity Selection I (Start Time, End Time)**

##### **19.3 – Activity Selection II (Deadline, Reward)**

##### **19.4 – Huffman encoding (Discussion)**

##### **19.5 – Huffman encoding (Algorithm Coding)**

#### 20. Knapsack problem

##### **20.1 – Previous Recap and Knapsack Problem**

##### **20.2 – Introduction to DP, Fibonacci number (using DP)**

##### **20.3 – 0 1 Knapsack Problem (Top- Down)**

##### **20.4 – 0 1 Knapsack Problem (Bottom-Up)**

##### **20.5 – 0 1 Knapsack Problem (Coding Bottom-Up)**

#### 21. Dynamic Programming, Subset sum problem, Equal sum partition

##### **21.1 – Dynamic Programming Recap**

##### **21.2 – Subset sum problem Top-Down Approach, Bottom-Up Approach**

##### **21.3 – Count of Subsets sum problem**

##### **21.4 – Equal Sum Partition**

##### **21.5 – Minimise Sum Difference Partition**

#### 22. knapsack problem, Subset sum, min number of ways, LCS of 2 strings

##### **22.1 – Coin Change Problem Number of ways**

##### **22.3 – Longest Common Subsequence**

#### 23. LCS Problem

##### **23.1 – LCS Length Recursive Approach**

##### **23.2 – LCS Length Iterative Approach**

##### **23.3 – Finding the LCS string**

##### **23.4 – Longest Common Substring**

##### **23.5 – Longest Palindromic Subsequence’s**

#### 24. Matrix chain multiplication, Palindrome partition, Minimum edit distance

##### **24.1 – Matrix Chain Multiplication**

##### **24.2 – Palindrome Partition**

##### **24.3 – Minimum Edit Distance**

##### **24.4 – Discussion on improvising DP and revision tips**

#### 25. Graph, Disjoint set, Union

##### **25.1 Previously done review**

##### **25.2 DSU Introduction**

##### **25.3 Introduction to Union and Find**

##### **25.4 Optimized Union and Find**

#### 26. Graph creation, BFS, DFS, cyclic acyclic graph, MST

##### **Graph creation, BFS, DFS, cyclic acyclic graph, MST**

##### **26.1 Graph Creation**

##### **26.2 BFS**

##### **26.3 DFS**

##### **26.4 Cyclic Acyclic Graphs**

##### **26.5 Kruskal Algorithm for MST**

#### 27. Prims algo MST, Shortest path, Topological Sort, Euler circuit Euler path

##### **27.1 Prims Algorithm for MST**

##### **27.2 Dijkstra Algorithm**

##### **27.3 Topological Sorting**

##### **27.4 LeetCode Problem Rotting Oranges**

##### **27.5 Euler Circuit, Euler Path**

##### **27.6 Dijkstra failure and other graph algorithm’s introduction**

#### 28. Bellman Ford, strongly connected component

##### **Bellman Ford, strongly connected component**

##### **28.1 Bellman Ford**

#### **Leetcode Video Solutions**

##### **1. Two Sum | Leetcode Easy.**

##### **2. Valid Sudoku | Leetcode Medium.**

##### **3. Group Anagrams | Leetcode Medium.**

##### **4. Container with most Water | Leetcode Medium.**

##### **5. Set Matrix Zeros | Leetcode Medium – Java.**

##### **5.1 Set Matrix Zeros | Leetcode Medium – Python Draft Lesson.**

##### **6. Median of two sorted arrays | Leetcode Hard.**

##### **7. Trapping Rain Water | Leetcode Hard – Python.**

##### **8. Best time to buy and sell stock | Leetcode Easy.**

##### **9. Best time to buy and sell stock ii | Leetcode Medium.**

##### **10. Product of Array Except Self | Leetcode Medium.**

##### **11. Find Minimum in Rotated Sorted array II | Leetcode Hard**

##### **12. Range Sum Query – Immutable | Leetcode Easy**

##### **13. Sort Colours | Leetcode Medium**

##### **14. Roman to Integer | Leetcode Easy**

##### **15. Candy Description | Leetcode Hard**

##### **16. Remove Duplicates from Sorted Arrays**

##### **17. Longest Substring without repeating characters | Leetcode Medium**

##### **18. Fruit into Baskets | Leetcode Medium**

##### **19. Text justification | Leetcode Hard**

##### **20. Plus One | Leetcode Easy**

##### **21. Subarray Sum Equal k | Leetcode Medium**

##### **22. Permutation in String | Leetcode Medium**

##### **23. Happy Number | Leetcode Easy**

##### **24. Reverse Linked List ii | Leetcode Medium**

##### **25. Design Linked List | Leetcode Medium**

##### **26. Palindrome Linked List | Leetcode Easy**

##### **27. Odd even Linked List | Leetcode Medium**

##### **28. Split Linked List in Parts | Leetcode Medium**

##### **29. Copy List with Random Pointer | Leetcode Medium**

##### **30. Reverse Linked List | Leetcode Easy**

##### **31. Linked List Random Node | Leetcode Medium**

##### **32. Swapping Nodes in a Linked List | Leetcode Medium**

##### **33. LRU Cache | Leetcode Medium**

##### **34. Remove Linked List Elements | Leetcode Easy**

##### **35. Flatten Nested List Iterator | Leetcode Medium**

##### **36. Remove Duplicates from Sorted List ii | Leetcode Medium**

##### **37. Convert Binary Number in a Linked List to Integer**

##### **38. Rotate List | Leetcode Medium**

##### **39. Insertion Sort List | Leetcode Medium: - Java.**

#### **Technical Mock Interview**

##### Technical Interview 1

##### Technical Interview 2

##### Technical Interview 3

##### Technical Interview 4

##### Technical Interview 5