



Benefits

- Logical Thinking**
- Analytical Skills**
- Computational Thinking Skills**
- Real-life problem solving Skills**
- Algorithmic Skills**
- Python Development**
- Codingal T-shirts and Goodies**
- Competitions to participate, build projects, test skills and win prizes**
- Webinars**
- Exclusive access to FunDay events**



Achievements

- Python Developer Certification**
- Algorithms Champion Certification**
- Coding Champion Certification [On course completion]**
- Badges and Leaderboard rankings**
- Lifetime community access**
- Mentoring session with Founder**
- Preparation for National & International competitions**

Modules	Theme	Topics Covered	Outcome
M1 (6 Classes)	Python basics 1	Students will learn the basics of the most popular language i.e. Python with the help of different activities.They will learn about data types, Conditionals and Loops,Functions.In addition they will also create patterns using turtle.	Students will learn about the python programming language including its history, applications, concepts like data types, conditionals and loops, patterns using turtle.
M2 (6 Classes)	Python basics 2	Students will learn about data structures, classes, polymorphism, encapsulation and object oriented programming and create various softwares using these concepts	Students will continue their journey in python by working on advanced topics i.e. classes, libraries, creating modules and importing them.
M3 (6 Classes)	Basics of Implementation and Mathematics	Space and time complexity, Input-Output, Count Digits, Palindrome Numbers, GCD and LCM, Check for prime, Prime factors of a number, Sieve of Eratosthenes	Students will work on different famous problems to build strong Mathematical Logic.
M4 (6 Classes)	Bitwise Operations	Representation of numbers as bits, Check if the number is odd using bits, Power of 2, Count set bits, Odd occurring element, Two odd occurring, Three odd occurring, kth bit set or not	Students will learn about bitwise operations in Python and how to use them to improve code efficiency.
M5 (6 Classes)	Recursion	Concept of Recursion with implementation, Problems on Recursion, Analysis of Recursion	Students will learn about recursion and it's complexity in Python and how to use it to improve your code's efficiency.



Modules	Theme	Topics Covered	Outcome
M6 (6 Classes)	Array	Operations(insert, delete) in array, The largest and smallest number in an array, Sum of elements, Reverse array, Move all 1's to right, Kadane's Algorithm, Pointer approach in array and more problems.	Students will learn about Array in Python, operations on arrays and important algorithms
M7 (6 Classes)	Searching and Sorting	Linear Search, Binary Search [Problems], Two Pointer Approach, Bubble Sort, Selection Sort	Students will learn about Searching and Sorting algorithms, concepts + implementation using Python and related problems
M8 (6 Classes)	Advanced Sorting Algorithms	Insertion Sort, Merge Sort , Quick Sort and related problems	Students will learn about Advanced Sorting algorithms, concepts + implementation using Python
M9 (6 Classes)	Linked List and Matrix	Linked List Concept, Implementation, Swapping Elements in Linked List, Two Pointed Linked List Techniques, Matrix , Introduction to Matrix, Matrix Transformation, Matrix Multiplication	Students will learn about nodes, the building-block data structure and Linked List using Python.
M10 (6 Classes)	Strings	Palindrome, Anagram, Basic Problem Solving in strings, Rabin Karp Algorithm, KMP Algorithm	Students will learn String concepts + implementation, searching algorithm and string problems using Python

Modules	Theme	Topics Covered	Outcome
M11 (6 Classes)	Stack and Queue	Stack Implementation with array, Implementation with Linked List, Balanced parenthesis , Double stack in an array, Queue, Implementation with array, Implementation with Linked List, Implement stack using queue	Students will learn about trees, how to create them and traverse them using Python
M12 (6 Classes)	Trees	Tree Introduction, Implementation, Application, Binary Tree, Tree Traversal : Inorder, Preorder, Postorder traversal	Students will learn about trees, how to create them and traverse them using Python
M13 (6 Classes)	BST, Binary Indexed Trees	Making BST, Search in BST, Insert in BST, Deletion in BST, Making segment trees, Use in DBMS, B+,B trees	Students will learn about Binary Search Trees, Binary Indexed trees and how to create them and traverse them.
M14 (6 Classes)	Heap, Hashing	Introduction to Heap, Implementation, Heapify, Heapsort, Priority queue, Introduction to Heap, Hash function, Collision handling, Questions regarding hashing	Students will learn about Heap, Priority queue, Hashmap, how to create them using Python and problems related to Heaps and Hashing
M15 (6 Classes)	Graphs	Graph using Adjacency matrix, Adjacency list, Breadth first search, Depth first search and related problems.	Students will learn about graphs and graph search in Python.

Modules	Theme	Topics Covered	Outcome
M16 (6 Classes)	Graph Algorithms, Backtracking	Dijkstra, Bellman ford, Floyd-Warshall Algorithm, N-queen, Sudoku Problem	Students will learn about learn about pathfinding algorithms and backtracking along with its implementation using Python
M17 (6 Classes)	Backtracking, Greedy Algorithm	Rat in a maze problem, Problems like Fractional Knapsack, Job Sequencing, Huffman Coding, Activity Selection Problem	Students will learn what are Learn about greedy algorithms and how to implement them in Python.
M18 (6 Classes)	Dynamic Programming 1	Introduction to Dynamic programming , Memoization, Tabulation, Factorial with DP vs Recursion, Longest Subsequence Problems ,Coin Change, Edit Distance Problem	Students will be introduced to Dynamic programming in Python. They will also work on some very challenging problems beginning from logic building to its implementation.
M19 (6 Classes)	Dynamic Programming 2	0-1 Knapsack, Subset Sum Problem , Maximum Sum with no consecutive, Minimum Coins to make a value, Minimum jumps to reach the end, Matrix Chain Multiplication, Palindromic Partitioning	Students will learn how to solve complex problems using Dynamic programming in Python.
M20 (6 Classes)	World of Competitive Programming	Solve Competitive programming questions asked in Coding competitions(Google Code Jam, Facebook Hacker Cup, etc.), Online rounds of top Product Based Companies(Google, Amazon, Facebook).	Students will learn how to solve complex Competitive programming problems using all the concepts we learned in the previous modules to crack competitive programming and coding contests.





Start coding with Codingal :)

It's fun and you can build anything you can imagine!

Any questions?
Contact me anytime.

Send a message

support@codingal.com

Call me

+91 63611 58952