CODING NINJAS

Competitive Programming

"Programmers are not to be measured by their ingenuity and their logic but by the completeness of their case analysis."

~Alan J. Perlis

About Coding Ninjas

At Coding Ninjas, our mission is to continuously innovate the best ways to train the next generation of developers and transform how tech education is delivered. Training is designed and provided by professional developers turned educators who have experience working at bigwigs like Facebook, Amazon, Google etc. and are Stanford, IIT, IIIT alumni.

Coding Ninjas teaches 17+ Programming courses in Foundation, Advanced, Data & Development courses such as Machine Learning, Data Science, Web Development, Android and more.

Doubt Support

We have developed a very scalable solution using which we are able to solve 4000+ doubts every single day with the help of 500+ doubts on the platform itself with an average rating of 4.8 out of 5.

Placement Cell

50,000+
Students taught so far

78%+
Percentage placement

2500+
Students placed in top MNCs

300+ Placement Partners | **7.6L** Average Salary
Number of placement partners and average salary of students

100+
Students received International job offers

Ankush Singla

Co-Founder & Instructor

Ankush holds a Bachelor's degree in Computer Science from India's most premier institute- IIT Delhi and a Master's degree in Computer Science from Stanford University.

He is a coding enthusiast and has worked with bigwigs like Amazon and Facebook in the past.

Live Mentor Support & Student Experience Team

Dedicated TAs and Student experience team to make sure that your doubts get resolved quickly and you don't miss your deadlines.

Want A Break? Pause Your Course

Take a short break when you need it. Pause your course for upto 60 days. Resume when you are ready

Get An Industry Recognised Certificate

Get awarded with an industry recognised certificate after you complete your programming course

Be A Part Of The Learning Community

Slack groups to meet your batchmates. Learn from your peers about resources, doubts and more!

Programme Overview

Course Overview

Learn to write the most efficient programmes and equip yourself to get solutions for the complex codes, for competitions like ACM-ICPC, Google Codejam and more. You will learn how to approach or solve a given problem in the most efficient way and practice challenging problems using the concepts of data structures and algorithms. Some of the advanced topics include Range queries, Number theory, Computational geometry etc..,

Features

60+
Hours of video content

180+
Coding problems

30+
Assignment

WHY
Competitive Programming

Programming skills enhance not just the codes but also one's critical thinking ability. Competitive programming makes you an expert in understanding and performing exceptional case analysis. From mobile applications to complex softwares all work on the code strings, programming is building cell for the organizations and opens career opportunities like Software Developer, Web Developer, Data Analysts etc. for companies like Google, Oracle, Amazon . This field is advancing everyday, hence the requirement of excellent programmers and analysts.

Companies Hiring

amazon | Adobe | Paytm

Google | zomato | ORACLE

Course Outcome

- You will learn how to approach or solve a given problem in the most efficient way and practice challenging problems using the concepts of data structures and algorithms. Some of the advanced topics include Range queries, Number theory, Computational geometry etc.
- This course will enhance concepts of data structures & algorithms and form the base for web development, mobile development or machine learning.

Placement after the course

Arpan Singh

ATLASSIAN

Vijay Kumar

Cognizant

Vinayak Dhiman

amazon

Samay Varshney

Microsoft

Tanya Prakash

Goldman Sachs

Ayush Raghuwanshi

L&T Infotech

Course Offerings

Basic	Standard	Pro	Premium
X	Introduction to Programming	X	Introduction to Programming
X	Data structures and algorithms	X	Data structures and algorithms
Competitive Programming	Competitive Programming	Competitive Programming	Competitive Programming
X	X	10 industry mentor sessions	10 industry mentor sessions
X	X	Resume Building Workshops	Resume Building Workshops
X	X	Help in profile building	Help in profile building
X	X	100+ curated interview problems	100+ curated interview problems
X	X	DSA mock test series to crack product companies	DSA mock test series to crack product companies
X	X	Premium Hirist Account	Premium Hirist Account
Dedicated Placement Cell with curated job	Dedicated Placement Cell with curated job	Dedicated Placement Cell with curated job	Dedicated Placement Cell with curated job
6 Months	12 Months	8 Months	14 Months
60+ Hours	120+ Hours	60+ Hours	120+ Hours
30+ Assignments	70+ Assignments	30+ Assignments	70+ Assignments
180+ Problems	500+ Problems	280+ Problems	600+ Problems

Introduction to Programming

Standard/Premium

Learn the basics of the most popular programming languages (C++ /Java / Python) and become an expert in the core fundamentals of programming.

Data Structures and Algorithms

Standard/Premium

Data structures and algorithms are all about organizing the information and finding the most efficient approach to solve a problem. Learning these concepts will in turn help you to improve your problem-solving skills and solve any real-world problems using technology.

Competitive Programming

Basic/Standard/Pro/Premium

You will learn how to approach or solve a given problem in the most efficient way and practice challenging problems using the concepts of data structures and algorithms. Some of the advanced topics include Range queries, Number theory, Computational geometry etc..,

Detailed Curriculum

Introduction to Programming

Topics	Sub-topics	Details
Problem Solving Techniques	Flowcharts	Introduction to flowcharts, Decision making using flowcharts, Loops, Example problems
	Variables and Data types	First program, Variables and data types, Taking input, How data is stored in memory, Arithmetic Operators
	Conditional statements	Introduction to If else, Relational and logical operators, Nested conditionals
Loops and Functions	While loops	While loops, Flow of execution of statements in while loop, Example problems using while loop
	Patterns	Introduction to patterns, Basic Patterns, Square Patterns, Triangular Patterns, Character Patterns, Reverse Triangle, Inverted patterns, Isosceles triangles
	For loops	For loops, Break and Continue, increment - decrement operators
	Functions	Introduction to functions, Working of function calling, Variables and its scope, Pass by value
Arrays	Introduction to arrays	Introduction to arrays, How arrays are stored in memory, Passing arrays to functions
	Searching and Sorting	Understanding Binary Search, Selection sort, Bubble sort, Insertion sort, Merging two sorted arrays
Strings and 2D Arrays	Strings	Introduction to strings, storage of strings and their inbuilt functions
	2D Arrays	2D arrays, Storage of 2D arrays, Example problems using 2D Arrays

Data Structures and Algorithms

Topics	Sub-topics	Details
Problem Solving Techniques	Recursion	Introduction to recursion, Principle of mathematical induction, Fibonacci numbers, Recursion using arrays, Recursion using strings, Recursion using 2D arrays
	Time and space complexity	Order complexity analysis, Theoretical complexity analysis, Time complexity analysis of searching and recursive algorithms, Theoretical space complexity, Space complexity analysis of merge sort
Object-oriented programming	Basics of OOP	Introduction to oops, Creating objects, Getters, and setters, Constructors and related concepts, Inbuilt constructor and destructor, Example classes
	Advance concepts of OOP	Static members, Function overloading and related concepts, Abstraction, Encapsulation, Inheritance, Polymorphism, Virtual functions, Abstract classes, Exception handling
Linear Data Structures	Linked lists	Introduction to linked list, Inserting node in linked list, Deleting node from linked list, Midpoint of linked list, Merge two sorted linked lists, merge sort of a linked list, Reversing a linked list
	Stacks and Queues	Introduction to stacks, Stack using arrays, Dynamic Stack class,Stack using linked list, Inbuilt stack, Queue using arrays, Dynamic queue class, Queue using linked list, Inbuilt queue
Trees	Generic Trees	Introduction to Trees, Making a tree node class, Taking a tree as input and printing, Tree traversals, Destructor for tree node class
	Binary Trees	Introduction to Binary Trees, Taking a binary tree as input and printing, Binary Tree traversals, Diameter of binary tree
	Binary Search Trees	Introduction to Binary Search Trees, Searching a node in BST, BST class, Inserting and Deleting nodes in BST, Types of balanced BSTs
Advanced Data Structures	Priority Queues	Introduction to Priority Queues, Ways to implement priority queues, Introduction to heaps, Introduction to Complete Binary Trees and its implementation, Insert and Delete operations in heaps, Implementing priority queues, Heap sort, Inbuilt Priority Queue

Topics	Sub-topics	Details
	Hashmaps	Introduction to Hashmaps, Inbuilt Hashmap, Hash functions, Collision handling, Insert and Delete operation implementation in hashmap, Load factor, Rehashing
	Tries	Introduction to Tries, Making a Trie Node class, Insert, Search and Remove operation implementation in Tries, Types of Tries, Huffman Coding
	Graphs	Introduction to Graphs, Graph Terminology, Graph implementation, Graph Traversals (DFS and BFS), Weighted and Directed Graphs, Minimum Spanning Trees, Cycle Detection in Graphs, Kruskal's algorithm, Prim's Algorithm, Dijkstra's algorithm
Dynamic Programming	Introduction to Dynamic Programming	Introduction to Memoization, Introduction to Dynamic Programming, Fibonacci numbers using recursion, memoization and dynamic programming
	Applications of Dynamic Programming	Longest Common Subsequence (LCS) using recursion, memoization and dynamic programming, Edit distance using recursion, memoization and dynamic programming, Knapsack problem using recursion, memoization and dynamic programming

Competitive Programming

Topics	Sub-topics	Details
Basics of Competitive Programming	Introduction to competitive programming	Why Competitive Programming?, How To Approach A Problem In Contest, Various Types Of Errors
	Basics of recursion	Introduction to Recursion, Recursion and PMI, Fibonacci Number, Recursion and Arrays
	Time and space complexity analysis	Order Complexity Analysis, Theoretical Analysis, Linear Search time complexity, Insertion Sort time complexity, Selection Sort time complexity, Theoretical Analysis - Recursive Algorithms, Merge Sort Time complexity, Fibonacci Time Complexity Analysis, Space Complexity Analysis, Merge Sort Space Complexity Analysis, Fibonacci Space Complexity Analysis, Kadane's Algorithm
	Language tools	STL - Data Structures, STL - Functions, Hussain Set, Voters List, Permutation & Palindrome
	searching and sorting applications	Aggressive Cows, Inversion Count
Applications of Recursion	Advanced Recursion	Recursion and Strings, Merge Sort Algorithm, Quick Sort Algorithm, Strings, Return Subsequences of a String, Return Keypad Approach
	BACKTRACKING	Backtracking (N-Queen), N-Queen Code, Rat In A Maze code
Bit Manipulation and Modulo Arithmetic	Bit manipulation	Flip ith bit, Check Odd-Even, Introduction & Shift Operators, Clear All Bits From LSB, Remaining Bitwise Operators, Check Power of 2, Check nth bit
	Modulo arithmetic	Modulo Operations, Modulo Properties, Modulo Operations - Continue, Number Of Balanced Binary Trees

Topics	Sub-topics	Details
Dynamic Programming	Dynamic programming	Basics Of Dynamic programming: Fibonacci Numbers, AlphaCode, Longest Increasing Subsequence, Coin Change And StairCase, Minimum Cost, Magic Grid, Maximum Sum Rectangle, Longest Common Subsequence, Knapsnack - Iterative, Subset Sum
	Greedy problems	Introduction To Greedy Techniques, Minimum Absolute Difference In Array, Nikunj And Donuts, Fractional Knapsack, Weighted Job Scheduling
	Dp and Bitmasking	DP With Bitmasking, What is Bit Masking?, Minimum Cost for Jobs, Mehta and Bank Robbery
Range Queries	Segment tree	Introduction to Segment Tree, Update On A Segment Tree, Size Of Segment Tree, Sum Of Squares Video, Maximum Sum in Subarray, Lazy Propagation, How To Build A Segment Tree, Query On A Segment Tree, Segment Tree Max Pair Sum
	Fenwick tree	Introduction to Fenwick Tree, Coordinate Compression, OrderSet Problem, Distinct Query
Graphs	Graph implementation	Return all connected components, Get Path - BFS, Get Path - DFS, Has path, Solution: BFS Traversal, DFS - Adjacency Matrix, Graphs Terminology, Graphs Introduction, Graphs Implementation, BFS Traversal, BFS & DFS for disconnected graph, Weighted & Directed Graphs
	Graph algorithms	Dijkstra’s Algorithm, Prim's Algorithm, Kruskal's Algorithm, Minimum Spanning Trees, Detect Cycle, Union Find Algorithm
	Advanced graphs	Permutation Swaps, Connected Components, Fill Matrix Question, Connected Horses, Strongly Connected Components, Bipartite Graph
Strings and Tries	String algorithms	Longest Prefix Suffix, Z-Algorithm Pattern Matching, Longest Palindromic Substring Basics, KMP Algorithm, Pattern Matching Basics, Longest Prefix Suffix

Topics	Sub-topics	Details
Tries	Dynamic programming	Search Engine, SUBXOR, Maximum XOR Subarra, XOR Maximum Pair, Tries & XOR
Mathematics for Competitive Programming	Number theory	GCD(Euclid Algorithm), Diophantine Equations, Multiplicative Modulo Inverse, Find Prime Numbers From 1 To N, Sieve Of Eratosthenes, Divisors Of Factorial, Advanced GCD, Euler's Totient Function, Segmented Sieve,Wilson's Theorem, Modular Exponentiation, Nth Fibonacci Number - Log(n), Fermat's Little Theorem, Matrix Exponentiation, Optimized Power Function, Recurrence Relations
	Game theory	Othello, Othello Evaluation Function, MinMax Algorithm, Sprague Grundy Theorem, Grundy Numbers, Game Of Nim, Intro To Game Theory, Proof Of Nim Formula, Evaluation Function In MinMax Algorithm
	Computational geometry	Convex Hull, Area Of A Polygon, Distance Of A Point, Intro To Computational Geometry, Intersection Of Two Lines

CODING NINJAS

1800-123-3598

contact@codingninjas.com

codingninjas.com

Follow us on