SIX PHRASE – The Finishing School

Prabhu N.D. – 99946 75750 | 962 962 0432
www.sixphrase.com | sixphrase@gmail.com | training@sixphrase.com
www.facebook.com/SixPhrase

QUOTATION

Sir/Madam,

Sub: Quotation for **Six Phrase – Talentely - PRIME - Placement Fit - Training & Hiring Program.**

- ***Training – 300 Hours of Technical, Aptitude & SoftSkills Training.***
- ***AI Enabled LMS (Learning Management System) & Talent Hiring Platform based on EMPLOYABILITY INDEX SCORE (EIS).***
- ***Hiring – Marquee (salary > 20lakhs per annum), Super Dream (salary 10 lakhs to 20lakhs per annum), Dream (salary 5 lakhs to 10lakhs per annum), and Service (salary < 5 lakhs per annum) Companies Placements.***
  - ***Opportunity 1:*** *During our training **program** enroll students in MySlate LMS. Follow up and motivate students to complete the courses in LMS. Award **reward points (EMPLOYABILITY INDEX SCORE)** for each problem/program a student solves in the LMS. Categorize Talent in 7th Semester based on scores accumulated **(EMPLOYABILITY INDEX SCORE)** in our LMS. Place students in Marquee, Super Dream, Dream & Services companies based on scores accumulated.*
    - ✓ ***Outcome based placements** based on below target **EMPLOYABILITY INDEX SCORE.***
      - ***Marquee Students >25000 points***
      - ***Super Dream Students > 20000 && < 25000 points***
      - ***Dream Students >15000 && <20000 points***
      - ***Service Students >12000 && <15000 points***
  - ***Opportunity 2:*** *Identify **Special Talent** in 7th Semester based on Employability Index Score accumulated in our LMS. Train the identified students on **Future Skills** (AI, ML, DS, Big Data, Cloud, Cyber Security, Full Stack etc…) and Deploy them. Students will **not be charged** for Future Skills Training.*
  - ***Opportunity 3:*** *Identify Talent who wish to **Upgrade** their existing service offer to product offer at the end of 7th or beginning of 8th Semester*

*(after Day1 placements). Train them on Future Skills and Deploy them. Students will **not be charged** for Future Skills Training.*

- ***Opportunity 4:*** *Identify **Unplaced & Deserving Talent** at the end of 7th or beginning of 8th Semester (after Day1 placements). Hire them with a **Stipend** of 15k to 20k per month and train the students for services companies and Deploy them.*

In line with the discussion we had with you we have here with detailed the program details along with the cost for conducting the **PRIME - Placement Fit - Training & Hiring Program**. Please find below the program details.

Program Duration:	PRIME - Prod Fit – 300 Hours / 50 Days – Aptitude & Technical Training Program PRIME - Code Fit – 300 Hours / 50 Days – Aptitude & Technical Training Program ✓ 6th Semester – 30 Days (each month 6 days for 5 months) & 7th Semester – 20 Days ✓ Company Specific Training will be complimentary
Target Audience:	BE Students – Third Year Students
Cost Per Student:	Rs. per student (exclusive of taxes) (**negotiable**)
Program Type:	**PRIME - PROD FIT (Product Batch Students):** **Aptitude – Level 3** **Technical – Basic & Advanced C, Basic & Advanced Data Structure, Algorithm** **PRIME - CODE FIT (Service Batch Students):** **Aptitude – Level 2** **Technical – Basic C, Basic Data Structure**
Target Outcome:	✓ Continuous Performance Monitoring using Six Phrase Learning Management System (**LMS**) - **http://sixphrase.com.** ✓ Outcome based training with the below mentioned target scores ○ **Marquee Students >25000 points** ○ **Super Dream Students > 20000 points** ○ **Dream Students >18000 points** ○ **Service Students >12000 points**

Online Test Portal:	One year access to 600+ Online Tests (Aptitude, Technical, **150+ Company Specific Tests)** using My Slate. **Technical:** Basic C – 330 Practice Programs & 15 Video Lectures Advanced C – 330 Practice Programs & 30 Video Lectures Basic DS – 300 Practice Programs & 61 Video Lectures Advanced DS – 300 Practice Programs & 30 Video Lectures Algorithms – 300 Practice Programs & 84 Video Lectures Java – 330 Practice Programs & 52 Video Lectures Python – 330 Practice Programs & 30 Video Lectures **Aptitude:** Quantitative Aptitude – 120 Tests & 35 Video Lectures Reasoning Aptitude – 120 Tests & 25 Video Lectures Verbal Aptitude – 90 Tests & 15 Video Lectures **Company Specific Tests:** 150+ Product & Service Companies Specific Tests & Video Lectures **Competitive Programming:** Google Code Jam – 10 Tests & 27 Video Lectures TCS CodeVita - 20 Tests & 27 Video Lectures Infy TQ - 20 Tests & 15 Video Lectures Hack with Infy - 10 Tests & 10 Video Lectures
Demo Login Details:	**www.talentely.com** (Built on Progressive Web App**)** Login ID – engg.student@vidh.ai Password – test@12345

Aptitude Syllabus

➢ ***QUANTITATIVE***

- ✓ Time And Work
- ✓ Time, Speed, Distance
- ✓ Averages
- ✓ Ratios & Proportions
- ✓ Sequence, Series And Progressions
- ✓ Co-Ordinate Geometry
- ✓ Statistics
- ✓ Combinatorics
- ✓ Linear Algebra
- ✓ Probability
- ✓ Percentages
- ✓ Permutations & Combinations
- ✓ Numbers
- ✓ Functions
- ✓ Mensuration
- ✓ Sudoku
- ✓ Percentage
- ✓ Trigonometry
- ✓ Set Theory
- ✓ Theory Of Equation
- ✓ Logarithm
- ✓ Functions And Graph
- ✓ Data Interpretation On Multiple Charts
- ✓ Profit And Loss
- ✓ Simple Equation
- ✓ Algebra

✓ Geometry
✓ Data Interpretion
✓ Alligations And Mixtures
✓ Simple Interest & Compound Interest
✓ Pipes And Cisterns
✓ Problems On Hcf And Lcm
✓ Areas, Shapes, Perimeter
✓ Height And Distance
✓ Partnership
✓ Races And Games
✓ Simplification
✓ Clocks & Calendars
✓ Problems On Ages
✓ Surds & Indices
✓ Data Sufficiency
✓ Logarithms
✓ Problems On Trains
✓ Cryptarithmetic
✓ Divisibility
✓ Numbers And Decimal Fractions
✓ Spatial Ability
✓ Chain Rule

➢ ***REASONING***

✓ Data Arrangements
✓ Number Series
✓ Lr – Arrangements
✓ Lr – Ranking
✓ Assertion And Reason
✓ Team Formations
✓ Conditional Syllogisms
✓ Statement And Conclusions
✓ Statement - Courses Of Action
✓ Syllogism
✓ Statement And Assumptions
✓ Critical Reasoning
✓ Coding And Decoding
✓ Odd Man Out
✓ Direction Sense
✓ Image Based Problems
✓ Blood Relationship
✓ Seating Arrangments
✓ Logical Deduction
✓ Character Puzzles
✓ Clock Puzzles
✓ Dot Situation
✓ Embedded Images
✓ Figure Matrix
✓ Grouping Of Images
✓ Image Analysis
✓ Logical Puzzles
✓ Mirror And Water Images
✓ Missing Letters Puzzles
✓ Number Puzzles
✓ Paper Cutting
✓ Paper Folding
✓ Pattern Completion
✓ Playing Cards Puzzles
✓ Rule Detection
✓ Shape Construction
✓ Attention To Details
✓ Flowcharts
✓ Puzzles
✓ Cubes
✓ Sequence And Series
✓ Statements
✓ Venn Diagrams
✓ Analogies
✓ Data Sufficiency
✓ Inferred Meaning
✓ Logical Order
✓ Mathematical Operations
✓ Logical Choice
✓ Analytical Reasoning

➢ ***VERBAL***

✓ Paragraph Formation
✓ Sentence Completion
✓ Reading Comprehensions
✓ Sentence Correction
✓ Spotting Errors
✓ Sentence Selection
✓ Antonyms
✓ Synonyms
✓ Jumbled Sentences
✓ Selecting Words
✓ Sentence Improvement
✓ Odd Words
✓ Sentence Formation
✓ Fact - Inference - Judgement

- ✓ Fill In The Blanks
- ✓ One Word Substitution
- ✓ Theme Detection
- ✓ Parts Of Speech
- ✓ Idioms And Phrases
- ✓ Spellings

SoftSkills Syllabus

- ✓ Dressing Etiquette
- ✓ Resume Writing Skills
- ✓ Group Discussion
- ✓ Interview Skills
- ✓ Mock Interview Sessions

Technical Syllabus

C, DataStructure & Algorithm Syllabus

Basic C Programming - Topics
C - Basic - Part 1 - Introduction to Programming
C - Basic - Part 2 - Data Types, Variables, Operators
C - Basic - Part 3 - Expressions, Precedence , Operators
C - Basic - Part 4 - Conditional Statements , Switch Statements
C - Basic - Part 5 - Looping
C - Basic - Part 6 - Digit Manipulation, Nested Loops, Patterns
C - Basic - Part 7 - Patterns , Number Problems
C - Basic - Part 8 - Array Basics
C - Basic - Part 9 - Structure
C - Basic - Part 10 - Pointers
C - Basic - Part 11 - Functions
C - Basic - Part 12 - Function Parameters
C - Basic - Part 13 - Abstract Data types, Array Operations
C - Basic - Part 14 - Time Complexity Analysis
C - Basic - Part 15 - Strings

Advanced C Programming - Topics
C - Advanced - Part 1 - Introduction to Bit Manipulation
C - Advanced - Part 2 - Problems on Bit Manipulation
C - Advanced - Part 3 - Introduction to Recursion
C - Advanced - Part 4 - Types of Recursion
C - Advanced - Part 5 - Solving Recurrence Relation I
C - Advanced - Part 6 - Solving Recurrence Relation II
C - Advanced - Part 7 - Solving Recurrence Relation III
C - Advanced - Part 8 - Time Complexity Analysis
C - Advanced - Part 9 - Indirect Recursion
C - Advanced - Part 10 – Solving Recurrence Relation for Indirect recursion and Nested

Recursion
C - Advanced - Part 11 - Tree Recursion
C - Advanced - Part 12 - Recursion - Sum of Natural Number
C - Advanced - Part 13 - Recursion - Factorial Number
C - Advanced - Part 14 - Recursion - Exponent/Power Function
C - Advanced - Part 15 - Recursion - Taylor Series
C - Advanced - Part 16 - Recursion - Taylor Series - Optimization Method 1
C - Advanced - Part 17 - Recursion - Taylor Series - Optimization Method 2
C - Advanced - Part 18 - Fibinocci Series
C - Advanced - Part 19 - Fibinocci Series - Optimization
C - Advanced - Part 20 - Combination Formula
C - Advanced - Part 21 - Tower of Hanoi
C - Advanced - Part 22 - Tower of Hanoi Implementation
C - Advanced - Part 23 - Introduction to Array
C - Advanced - Part 24 - Static vs Dynamic Array
C - Advanced - Part 25 - Two Dimensional Matrix
C - Advanced - Part 26 - Arrays in Compilers
C - Advanced - Part 27 - Matrix Representation - Row-Major , Column-Major
C - Advanced - Part 28 - Array ADT - display , append, insert, delete
C - Advanced - Part 29 - Array ADT - delete, get, set, search
C - Advanced - Part 30 - Array ADT - shift / rotate
C - Advanced - Part 31 - Inserting in a sorted Arrayand checking if an Array is sorted
C - Advanced - Part 32 - Segregating positive and negative numbers
C - Advanced - Part 33 - Merging two sorted arrays
C - Advanced - Part 34 - Set Operations - Union, Intersection
C - Advanced - Part 35 - Set Operations - Difference, Set Membership
C - Advanced - Part 36 - Finding missing elements in Arrays - Different Methods
C - Advanced - Part 37 - Finding duplicate elements in a Integer array
C - Advanced - Part 38 - Check for Anagrams in a String and Permutations of a String
C - Advanced - Part 39 - Matrices - Diagonal Matrix
C - Advanced - Part 40 - Matrices - Lower Triangular Matrix
C - Advanced - Part 41 - Matrices - Upper Triangular Matrix
C - Advanced - Part 42 - Matrices - Symmetric Matrix
C - Advanced - Part 43 - Matrices - Tri-diagonal Matrix
C - Advanced - Part 44 - Matrices - Square Band Matrix
C - Advanced - Part 45 - Matrices - Toeplitz Matrix
C - Advanced - Part 46 - Sparse Matrix - Introduction
C - Advanced - Part 47 - Addition of Sparse Matrices
C - Advanced - Part 48 - Polynomial Representation and Evaluation
C - Advanced - Part 49 - Polynomial Addition
Basic Data Structure Programming - Topics
Part 1 - Why Dynamic data structure ?
Part 2 - Introduction to Linked List
Part 3 - More about Linked List and Displaying a Linked List
Part 4 - Counting all the Nodes in a Linked list and Sum of all elements in the Linked List

Part 5 - Finding Maximum in a Linked list and Searching in a Linked list
Part 6 - Improving Searching in Linked list
Part 7 - Inserting in a Linked list
Part 8 - Inserting in a sorted linked list and Deleting from a linked list
Part 9 - Checking if a Linked list is sorted
Part 10 - Removing duplicates from a Linked List
Part 11 - Reversing Linked List
Part 12 - Reversing Linked List using Sliding Pointers and recursive technique
Part 13 - Concatenating and Merging two Linked list
Part 14 - Check for Loop in Linked List
Part 15 - Circular Linked List
Part 16 - Insert and Display in a Circular Linked list
Part 17 - Deleting from a Circular Linked list
Part 18 - Doubly Linked List - Insertion
Part 19 - Doubly Linked List - Deletion
Part 20 - Doubly Linked List - Reverse
Part 21 - Circular Doubly Linked List
Part 22 - Comparison of Linked list
Part 23 - Comparison of Linked List and Array
Part 24 - Finding Middle element of a linked list and Intersecting point of Two Linked list
Part 25 - Sparse Matrix using Linked list
Part 26 - Polynomial using Linked list
Part 27 - Introduction to Stack and Stack using Array
Part 28 - Stack using Linked list
Part 29 - Paranthesis Matching
Part 30 - Infix to Postfix Conversion
Part 31 - Associativity and unary Operator
Part 32 - Infix to Postfix using Stack method 1
Part 33 - Infix to Postfix using Stack method 2
Part 34 - Introduction to Queue ADT
Part 35 - Queue using Single Pointer and Two Pointer
Part 36 - Queue using Array and it's drawback
Part 37 - Circular Queue
Part 38 - Queue using Linked list
Part 39 - Double ended queue - Dequeue
Part 40 - Priority Queue
Part 41 - Queue using two Stacks

Advanced Data Structure Programming – Topics (Prod Fit)

Part 1 - Trees (Terminology)
Part 2 - Number of Binary Trees using N Nodes
Part 3 - Height vs Nodes in Binary Tree
Part 4 - Internal Nodes Vs External Nodes in Binary Tree
Part 5 - Strict Binary Tree and Height vs Node of strict Binary Tree
Part 6 - Internal vs External Nodes of Strict Binary Tree

Part 7 - n-ary Trees
Part 8 - Analysis of n-ary Trees
Part 9 - Representation of binary Tree
Part 10 - Linked Representation of Binary Tree
Part 11 - Full vs Complete Binary Tree
Part 12 - Strict vs Complete Binary Tree
Part 13 - Binary Tree Traversals - method 1
Part 14 - Binary Tree Traversals - method 2 and method 3
Part 15 - Creating Binary Tree
Part 16 - Preorder Tree Traversal
Part 17 - Inorder Tree Traversal
Part 18 - Iterative Preorder and Inorder
Part 19 - Level Order Traversal
Part 20- Generating Tree from Traversal
Part 21 - Height and count of Binary Tree
Part 22 - Count Leaf Nodes of a Binary Tree
Part 23 - Introduction to Binary Search Tree
Part 24 - Searching in Binary Search Tree
Part 25 - Inserting in a Binary Search Tree - Iterative
Part 26 - Inserting in a Binary Search Tree - Recursive
Part 27 - Creating Binary Search Tree
Part 28 - Deleting from Binary Search Tree
Part 29 - Generating BST from Preorder
Part 30 - Drawbacks of Binary Search Tree
Part 31 - Introduction to AVL Trees
Part 32 - Inserting in AVL with Rotations
Part 33 - General form of AVL Rotation
Part 34 - Generating AVL Tree
Part 35 - Deletion from AVL Tree with Rotations
Part 36 - Height Analysis of AVL Tree
Part 37 - 2-3 Trees
Part 38 - 2-3-4 Trees
Part 39 - Red Black Tree - Introduction
Part 40 - Red Black Tree Creation
Part 41 - Red Black Tree vs 2-3-4 Trees
Part 42 - Red Black Tree - Deletion
Part 43 - Introduction to Heap
Part 44 - Inserting in a Heap
Part 45 - Creating a Heap
Part 46 - Deleting from Heap and Heap Sort
Part 47 - Heapify - Faster Method for creating Heap
Part 48 - Heaps as Priority Queue
Part 49 - Introduction to Graphs
Part 50 - Representation of Undirected Graphs
Part 51 - Representation of Directed Graph
Part 52 - Breadth First Search

Part 53 - Depth First Search

Part 54 - Spanning Trees

Part 55 - Prim's Minimum Spanning Tree

Part 56 - Kruskal's Minimum Spanning Tree

Part 57 - Disjoint Subsets

Part 58 - Asymptotic Notations Big oh, Omega, Theta

Algorithms (Prod Fit)

Part 1 – Introduction to Algorithms and Backtracking

Part 2 – N Queen Problem

Part 3 – Knight's Tour Problem

Part 4 – Rat in a Maze Problem

Part 5 – Subset Sum Problem

Part 6 – Graph Coloring Problem

Part 7 – Hamiltonian Cycle

Part 8 – Sudoku Solver

Part 9 – Prime Numbers after Prime P with sum S

Part 10 – Permutations of a given String

Part 11 – Print all possible paths from top left to bottom right of a m x n matrix.

Part 12 – Introduction to Divide and Conquer

Part 13 – Binary Search explored

Part 14 – Merge Sort

Part 15 – Quick Sort

Part 16 – Strassen's Matrix Multiplication

Part 17 – Introduction to Dynamic Programming

Part 18 – Longest Common Subsequence

Part 19 – Longest Palindromic Subsequence

Part 20 – 0/1 Knapsack Problem

Part 21 – Subset Sum Problem

Part 22 – Minimum Cost Path

Part 23 – Coin Change Problem

Part 24 – Kadane's Algorithm

Part 25 – Minimum Edit Distance

Part 26 – Longest Increasing Subsequence

Part 27 – Introduction to Greedy Algorithm

Part 28 – Activity Selection Problem

Part 29 – Kruskal's Minimum Spanning Tree

Part 30 – Prim's Minimum Spanning Tree

Part 31 – Boruvka's Minimum Spanning Tree

Part 32 – Dijkstra's Shortest Path Algorithm

Part 33 – Minimum cost to connect all cities

Part 34 – Introduction to Pattern Searching

Part 35 – Naive Pattern Searching

Part 36 – KMP Algorithm

Part 37 – Rabin-Karp Algorithm

Part 38 – Boyer Moore Algorithm – Bad Character Heuristic