Complete Guide to HackWithInfy 2025

🚀 Introduction

HackWithInfy is a national-level coding competition conducted by **Infosys** to identify and recruit the **best coding talent** in India. It serves as an opportunity for programmers to showcase their problem-solving abilities and secure technical roles at **Infosys**.

🎯 Roles & Compensation Offered Through HackWithInfy:

Role	Compensation (CTC)	Selection Criteria
System Engineer (SE)	₹3.5 LPA	Basic coding proficiency
Digital Specialist Programmer (DSP)	₹6.2 LPA	Strong DSA & problem-solving
Specialist Programmer (SP)	₹9.5 LPA	Advanced DSA & competitive coding

🏆 HackWithInfy Competition Structure

🔹 Round 1: Online Coding Challenge

 Duration: 3 hours
 Format: 3 coding problems of varying difficulty
 Selection Criteria: Higher accuracy and efficiency in solving problems
 Skills Tested: DSA, Algorithmic Thinking, Optimization, Coding Speed

 Practice Platforms for Round 1:

- GeeksforGeeks Practice
- LeetCode Problem Set
- HackerRank Interview Prep
- CodeChef Contests

 Round 2: Grand Finale (Hackathon)

 Duration: 48 hours
 Format: Team-based hackathon
 Objective: Build an innovative project on real-world problems
 Technologies Covered: Web Development, AI/ML, Cloud Computing, Blockchain, IoT
 Selection Criteria: Code quality, innovation, feasibility

 Hackathon Preparation Resources:

- Infosys Hackathon Guide

- Build a Full-Stack Project (YouTube)
- GitHub Awesome Hackathons

DSA Syllabus for HackWithInfy

Basic Topics (For System Engineer Role - SE, ₹3.5 LPA)

- **Arrays & Strings:** Two Pointers, Sliding Window, Sorting
- **Sorting & Searching:** Binary Search, QuickSort, MergeSort
- **Recursion & Backtracking:** Subsets, N-Queens, Sudoku Solver
- **Basic Graph Algorithms:** BFS, DFS

Resources for SE Role:

- Striver's DSA Sheet (Basic)
- GeeksforGeeks Sorting & Searching

Intermediate Topics (For Digital Specialist Programmer - DSP, ₹6.2 LPA)

- **Dynamic Programming:** 0/1 Knapsack, Longest Common Subsequence

- **Advanced Graphs:** Dijkstra's Algorithm, Floyd Warshall
- **Trie & Hashing:** Prefix Trees, Anagrams
- **Greedy Algorithms:** Huffman Encoding, Activity Selection

Resources for DSP Role:

- LeetCode Medium Problems
- Dynamic Programming (YouTube)

Advanced Topics (For Specialist Programmer - SP, ₹9.5 LPA)

- **Segment Trees & Fenwick Tree**
- **Heavy-Light Decomposition (HLD)**
- **Bit Manipulation & XOR Tricks**
- **Game Theory & Mathematical Algorithms**

Resources for SP Role:

- CodeForces 1500+ Problems
- Advanced CP Guide

Preparation Strategy for HackWithInfy

Step 1: Understanding the Pattern

- Analyze previous year problems (Links provided below)
- Identify key topics that Infosys focuses on

Step 2: Practice Problems Daily

- **LeetCode (50 Easy + 100 Medium + 50 Hard)**
- **CodeChef Long Challenges & CodeForces Contests**

Step 3: Mock Tests & Time Management

- Solve **3-hour mock tests** weekly
- Improve speed and accuracy

Step 4: Participate in Contests

- **Weekly CodeForces & LeetCode Contests**
- **Join Infosys Hackathons for Experience**

Previous Year HackWithInfy Questions

HackWithInfy 2023 Previous Year Questions:

- HackWithInfy 2023 Questions PDF

HackWithInfy 2022 Previous Year Questions:

- HackWithInfy 2022 Questions PDF

HackWithInfy 2021 Previous Year Questions:

- HackWithInfy 2021 Questions PDF

Additional Resources

Resource	Link
HackWithInfy Official Page	Infosys Careers – HackWithInfy
GeeksforGeeks Interview Prep	GeeksforGeeks DSA Guide
Infosys Preparatory Guidance	Infosys Preparation Portal
LeetCode Premium for Practice	LeetCode Premium
CodeChef Infosys Contests	CodeChef Practice

Final Tips to Crack HackWithInfy

For SE Role: Focus on **basic problem-solving, sorting, recursion, and simple DSA concepts.**
For DSP Role: Practice medium-level DP, graphs, and backtracking. Improve speed.
For SP Role: Compete regularly in **CodeForces (1500+), LeetCode Hard, and AtCoder contests.**
During Contests: Optimize solutions, reduce time complexity, and handle edge cases.
For the Hackathon: Learn **full-stack development, AI/ML, APIs, and cloud computing.**

Need More Help?

Join Infosys HackWithInfy Discussion Groups:

- Infosys HackWithInfy Telegram Group
- Infosys HackWithInfy Discord

Top 50 Questions For Hackwithinfy

1. Arrays and Strings

- Find the maximum subarray sum (Kadane's Algorithm)
- Merge two sorted arrays without extra space
- Find the longest substring without repeating characters
- Rotate an array by 'k' positions
- Check if a string is a valid palindrome

2. Linked Lists

- Reverse a linked list
- Detect and remove a cycle in a linked list
- Find the intersection point of two linked lists
- Merge two sorted linked lists
- Remove the nth node from the end of a linked list

3. Trees and Graphs

- Perform level-order traversal of a binary tree
- Check if a binary tree is a valid binary search tree
- Find the lowest common ancestor in a binary tree
- Implement depth-first search (DFS) and breadth-first search (BFS)
- Detect a cycle in an undirected graph

4. Dynamic Programming

- Solve the 0/1 Knapsack problem
- Find the longest increasing subsequence
- Compute the edit distance between two strings
- Partition a set into two subsets with minimum difference
- Calculate the number of ways to climb stairs with 'n' steps

5. Backtracking

- Solve the N-Queens problem
- Find all permutations of a given string
- Generate all subsets of a set
- Solve the Sudoku puzzle

- Find all unique combinations that sum up to a target value

6. Sorting and Searching

- Implement quicksort and mergesort algorithms
- Find the kth largest element in an array
- Search in a rotated sorted array
- Find the first and last position of an element in a sorted array
- Perform binary search on a sorted array

7. Greedy Algorithms

- Find the minimum number of coins for a given amount
- Schedule maximum number of activities that don't overlap
- Find the Huffman coding for data compression
- Determine the minimum spanning tree using Kruskal's algorithm
- Solve the fractional knapsack problem

8. Recursion

- Generate all valid parentheses combinations for 'n' pairs
- Calculate the power of a number using recursion

- Solve the Tower of Hanoi problem
- Find the nth Fibonacci number using recursion
- Compute the greatest common divisor (GCD) of two numbers

9. Bit Manipulation

- Find the single number in an array where every other element appears twice
- Count the number of 1 bits in an integer
- Determine if a number is a power of two
- Reverse the bits of an integer
- Find the two non-repeating elements in an array where every other element repeats twice

10. Miscellaneous

- Design a LRU (Least Recently Used) cache
- Implement a trie (prefix tree)
- Evaluate the expression of a Reverse Polish Notation
- Find the median of two sorted arrays
- Implement an algorithm to serialize and deserialize a binary tree

---

Preparation Resources:

- **GeeksforGeeks:** A comprehensive platform offering tutorials and practice problems on various data structures and algorithms.
    - GeeksforGeeks Practice
- **LeetCode:** A popular platform with a vast collection of coding problems categorized by difficulty and topic.
    - LeetCode Problem Set
- **HackerRank:** Offers a variety of coding challenges and competitions to enhance problem-solving skills.
    - HackerRank Algorithms
- **CodeChef:** Provides competitive programming contests and a rich problem archive.
    - CodeChef Practice Problems

Additional Tips:

- **Understand the Concepts:** Before jumping into problem-solving, ensure you have a solid grasp of the underlying concepts.
- **Practice Regularly:** Consistency is key. Dedicate time daily to solve problems and review solutions.

- **Analyze Previous Year Questions:** Reviewing past HackWithInfy questions can provide insights into the types of problems asked.
- **Participate in Mock Tests:** Simulate the exam environment by taking timed mock tests to improve speed and accuracy.

Here are the important resource links for HackWithInfy preparation:

1. **GeeksforGeeks Practice:** https://practice.geeksforgeeks.org/
2. **LeetCode Problem Set:** https://leetcode.com/problemset/all/
3. **HackerRank Algorithms:** https://www.hackerrank.com/domains/algorithms
4. **CodeChef Practice Problems:** https://www.codechef.com/practice
5. **Infosys HackWithInfy Page:** https://www.infosys.com/careers/hackwithinfy.html
6. **Codeforces Competitive Programming:** https://codeforces.com/

7. **HackWithInfy Previous Year Questions (GeeksforGeeks):** https://www.geeksforgeeks.org/hackwithinfy-previous-year-questions/
8. **InterviewBit Coding Practice:** https://www.interviewbit.com/coding-interview-questions/
9. **TopCoder Competitive Programming:** https://www.topcoder.com/
10. **AtCoder for Competitive Programming:** https://atcoder.jp/

Role Specific DSA Requirement

1. System Engineer (SE) – ₹3.5 LPA

DSA Requirement: Basic to Intermediate

- **Focus Areas:**
  - Arrays & Strings
  - Sorting (Bubble Sort, Quick Sort, Merge Sort)
  - Searching (Binary Search, Linear Search)
  - Stacks & Queues (Basic Implementation)
  - Recursion (Basic Problems)
  - Simple Greedy Algorithms

    - Basic Hashing
- **Difficulty Level:**
    - **Easy to Medium** questions on platforms like **HackerRank, LeetCode (Easy), and GeeksforGeeks**
    - Usually 1-2 easy DSA problems in **HackWithInfy Round 1** will be enough for selection.
- **Resources to Prepare:**
    - GeeksforGeeks DSA Sheet (Basic)
    - LeetCode Easy Problems
    - Sorting and Searching Playlist (YouTube)

---

2. Digital Specialist Programmer (DSP) – ₹6.2 LPA

DSA Requirement: Intermediate to Advanced

- **Focus Areas:**
    - Everything from SE, plus:
    - **Graph Algorithms** (BFS, DFS, Dijkstra's Algorithm, Floyd-Warshall)
    - **Dynamic Programming (DP)** (0/1 Knapsack, Longest Common Subsequence)
    - **Backtracking** (N-Queens, Sudoku Solver)

    - **Trie (Prefix Tree)** (Autocomplete, Word Dictionary)
    - **Advanced Greedy Algorithms**
- **Difficulty Level:**
    - **Medium to Hard** problems on **LeetCode, CodeChef (Div2), CodeForces (1200-1400 rating)**
    - Expect 2-3 **medium difficulty problems** in **HackWithInfy Round 1** to qualify for DSP.
- **Resources to Prepare:**
    - Striver's DSA Sheet (Best for Intermediate-Level Prep)
    - LeetCode Medium Problems
    - Dynamic Programming Playlist (YouTube)

---

3. Specialist Programmer (SP) – ₹9.5 LPA

DSA Requirement: Advanced to Competitive Programming Level

- **Focus Areas:**
    - Everything from SE and DSP, plus:
    - **Segment Trees & Fenwick Tree (Binary Indexed Tree)**

    - **Heavy-Light Decomposition (HLD)** (for trees)
    - **Advanced Dynamic Programming** (Digit DP, Matrix Exponentiation)
    - **Bit Manipulation** (XOR Tricks, Bitmasking)
    - **Game Theory**
    - **String Matching Algorithms** (KMP, Z Algorithm)
- **Difficulty Level:**
    - **Hard** problems on **LeetCode, CodeForces (1500+ rating), AtCoder, CodeChef Div1**
    - In **HackWithInfy Round 1**, solving **at least 2 hard-level DSA problems** is required for SP selection.
- **Resources to Prepare:**
    - Advanced CP Sheet by Errichto (Google Engineer)
    - CodeForces 1500+ Rating Problems
    - Advanced DP Problems (YouTube)

Summary Table of DSA Requirement Per Role

Role	DSA Level Required	Key Topics
System Engineer (SE)	Basic to Intermediate	Sorting, Searching, Stacks & Queues, Simple Greedy, Basic Recursion
Digital Specialist Programmer (DSP)	Intermediate to Advanced	Graphs (BFS, DFS), Dynamic Programming, Backtracking, Trie, Advanced Greedy
Specialist Programmer (SP)	Advanced to Competitive Programming	Segment Trees, HLD, Advanced DP, Bit Manipulation, Game Theory, String Matching

Final Advice:

- **For SE:** Focus on **basic DSA** and ensure speed in problem-solving.
- **For DSP:** Focus on **graph theory, DP, and advanced recursion** techniques.

- **For SP:** You must be comfortable with **competitive programming, solving CodeForces 1500+ rated problems daily**.