



# HackWithInfy Preparatory Guidance



# INFOSYS ONLINE TEST

## TEST FORMAT

Test usually contains 3 questions that will have to be attempted in 3 hours

Test will have a sectional cut-off as well as a total cut-off

Each question will belong to different difficulty level:

**Easy** – Simple questions that can be solved by basic application of aptitude, algorithm and data structures

**Medium** – Usually a question based on Greedy algorithm

**Hard** – Usually a question based on Dynamic Programming

Code can be written for each question using one of the programming language from C/C++/Java/Python/JavaScript

# INFOSYS ONLINE TEST – HOW TO PREPARE

## DIFFICULTY OF QUESTION - MEDIUM

A **Greedy algorithm** is any algorithm that follows the problem-solving heuristic of making the locally optimal choice at each stage

Types of Greedy algorithm:

1. Pure greedy algorithms
2. Orthogonal greedy algorithms
3. Relaxed greedy algorithms

# INFOSYS ONLINE TEST – HOW TO PREPARE

## DIFFICULTY OF QUESTION - HARD

**Dynamic Programming** (DP) is an algorithmic technique for solving an optimization problem by breaking it down into simpler subproblems and utilizing the fact that the optimal solution to the overall problem depends upon the optimal solution to its subproblems.

### Principles of Dynamic Programming:

- Breaking the problem down into subproblems and calculating their values. Next time, upon encountering the same subproblem, the value can be reused instead of recalculation
- Avoid repeated work by remembering partial results, a common approach for performance enhancement

# INFOSYS ONLINE TEST – HOW TO PREPARE

## DIFFICULTY OF QUESTION - HARD

### Characteristics of a Dynamic Programming problem

Following are the two main properties of a problem that suggest the given problem can be solved using Dynamic Programming:

1. Overlapping subproblems

A problem has overlapping sub-problems if its solution involves solving the same subproblem multiple times.

2. Optimal substructure properties

The overall optimal solution can be constructed from the optimal solutions of its subproblems.

# INFOSYS ONLINE TEST – HOW TO PREPARE

## DIFFICULTY OF QUESTION - HARD

### Dynamic Programming methods

There are two different ways to store values for an **overlapping subproblem** so that the values can be reused:

#### 1. Top-down with Memorization

To solve a bigger problem, in this approach we recursively find the solution to the smaller sub-problems and its result is cached. So next time the same sub-problem is tried to solve, the cached memorized result is returned.

#### 2. Bottom-up with Tabulation

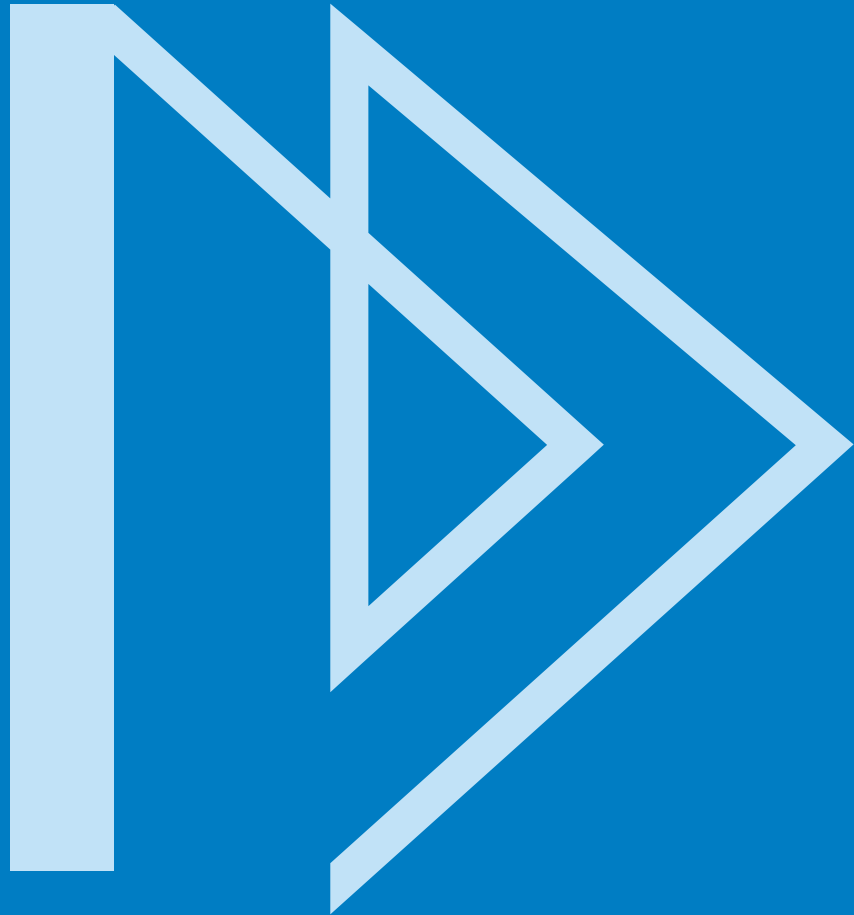
In this approach, we solve the problem “bottoms-up” i.e., by filling up an n-dimensional table. Based on the results in the table, the solution to the top/original problem is then computed.

# INFOSYS ONLINE TEST – HOW TO PREPARE

## DIFFICULTY OF QUESTION - HARD

### Some examples of Dynamic Programming

- Knapsack problem
- Fibonacci Numbers
- Palindromic Subsequence
- Longest Common Substring
- Dijkstra's algorithm



**THANK YOU**