

**Department of Computer Science & Engineering**  
**Motilal Nehru National Institute of Technology Allahabad**  
**Analysis of Algorithm (CSN13400) - External Lab Examination**

**Question 1: Priority Queue**

**Statement:** Given an array of strings words and an integer k, return the k most frequent strings.

**Sample Test Cases:**

Input 1: *words* = ["i", "love", "leetcode", "i", "love", "coding"], *k* = 2

Output 1: ["i", "love"]

Input 2: *words* = ["the", "day", "is", "sunny", "the", "the", "the", "sunny", "is", "is"], *k* = 4

Output 2: ["the", "is", "sunny", "day"]

**Question 2: Binary Tree**

**Statement:** Given an integer n, return the number of structurally unique BST's (binary search trees) which has exactly n nodes of unique values from 1 to n.

**Sample Test Cases:**

Input 1: [*n* = 3]

Output 1: 5

Input 2: [*n* = 1]

Output 2: 1

Input 3: [*n* = 5]

Output 3: 42

Input 4: [*n* = 9]

Output 4: 4862

### Question 3: Dynamic Programming

**Statement:** You are climbing a staircase. It takes  $n$  steps to reach the top.

Each time, you can climb either 1 or 2 steps. In how many distinct ways can you climb to the top?

Implement this using the concept of Dynamic Programming

**Sample Test Cases:**

Input 1:  $[n = 2]$

Output 1: 2

Input 2:  $[n = 3]$

Output 2: 3

Input 3:  $[n = 17]$

Output 3: 2584

Input 4:  $[n = 35]$

Output 4: 14930352

### Question 4: Graph

**Statement:** Given an adjacency list for an undirected graph, return true if and only if it is bipartite.

**Sample Test Cases:**

**Example 1:**

**Input:**  $graph = [[1, 2, 3], [0, 2], [0, 1, 3], [0, 2]]$

**Output:** *false*

**Example 2:**

**Input:**  $graph = [[1, 3], [0, 2], [1, 3], [0, 2]]$

**Output:** *true*

**Example 3:**

**Input:**  $graph = [[4, 3, 1], [3, 2, 4, 0], [1, 3], [0, 1, 2, 4], [0, 1, 3]]$

**Output:** *false*