Algorithms

DFS Homework 1

Mostafa S. Ibrahim
Teaching, Training and Coaching for more than a decade!

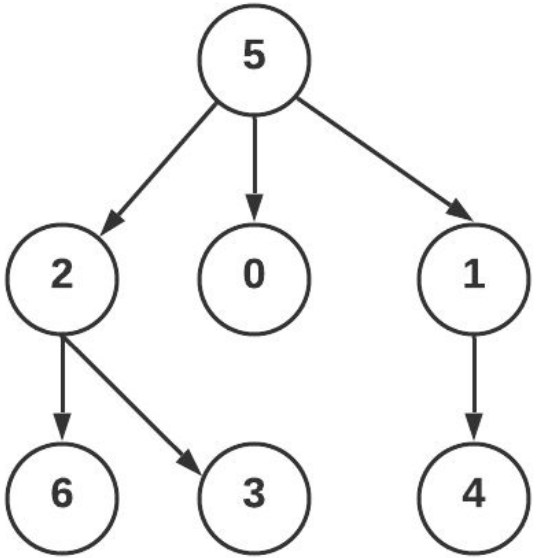
Artificial Intelligence & Computer Vision Researcher
PhD from Simon Fraser University - Canada
Bachelor / Msc from Cairo University - Egypt
Ex-(Software Engineer / ICPC World Finalist)

Problem #1: LeetCode 582 - Kill Process

- Given a **rooted tree**, and a specific node in the tree, return all the nodes that are below this tree node, including the node itself
- Context: these are **processes** and each process may have a parent process. If we killed a process, it kills all its children
  - In OS: we can have 10 processes, but their IDs are huge (13454, 454232, 123214, etc)
- Please **read** the website problem statement
- vector<int> killProcess(vector<int> &pid, vector<int> &ppid, int kill)
  - pid and ppid are the directed edges. Ith edge is: (ppid[i], pid[i])
  - **1 <= n <= 50000, 1 <= pid[i] <=50000, 0 <= ppid[i] <= 50000**
  - Only one process has no parent. Its ppid[root] = 0
  - All the values of pid are **unique**. The kill node is guaranteed to be in pid.
  - The returned values are in any order

My example

- Root here is 5
  - ppid(5) = 0
- ppid(2) = 5, ppid(4) = 1
- kill(5) ⇒ Gives all the **rooted** tree nodes
- kill(2) = {2, 3, 6}
- kill(4) = {4}



Problem #2: LeetCode 690 - Employee Importance

You have a data structure of employee information, which includes the employee's unique ID, their importance value, and their direct subordinates' IDs.

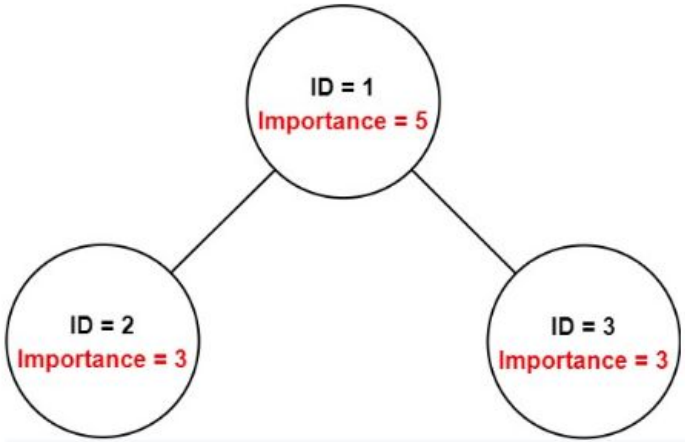
You are given an array of employees `employees` where:

- `employees[i].id` is the ID of the `i`th employee.
- `employees[i].importance` is the importance value of the `i`th employee.
- `employees[i].subordinates` is a list of the IDs of the direct subordinates of the `i`th employee.

Given an integer `id` that represents the ID of an employee, return *the* ***total*** *importance value of this employee and all their direct subordinates*.

- int getImportance(vector<Employee*> employees, int id)
  - Work directly on the given **rooted tree**. Don't convert to a standard graph
  - Return the sum of the nodes' values that are below this tree node, including the node itself

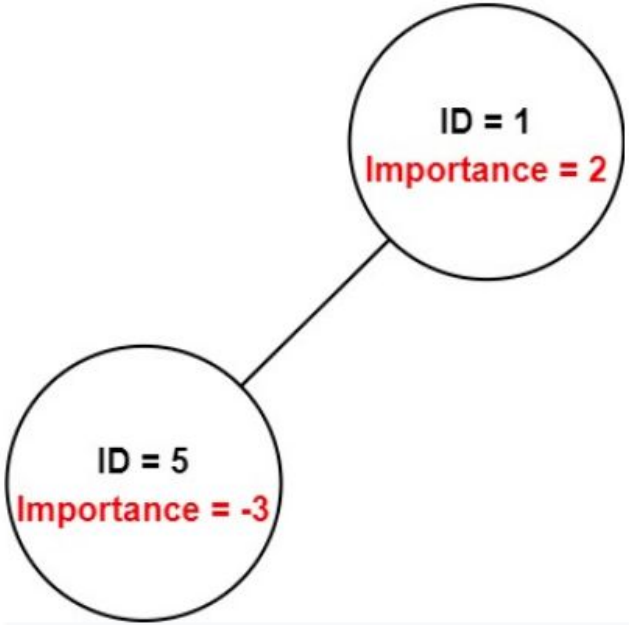
Example 1:



Input: employees = [[1,5,[2,3]],[2,3,[]],[3,3,[]]], id = 1
Output: 11
Explanation: Employee 1 has an importance value of 5 and has two direct subordinates: employee 2 and employee 3.
They both have an importance value of 3.
Thus, the total importance value of employee 1 is 5 + 3 + 3 = 11.

Note: if there are more nodes under ID=3, they are also summed to the totoal

Example 2:

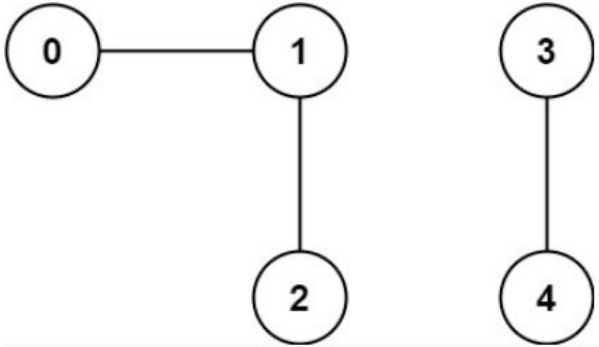


Input: employees = [[1,2,[5]],[5,-3,[]]], id = 5
Output: -3
Explanation: Employee 5 has an importance value of -3 and has no direct subordinates.
Thus, the total importance value of employee 5 is -3.

Problem #3: LeetCode 323 - Number of Connected Components in an Undirected Graph

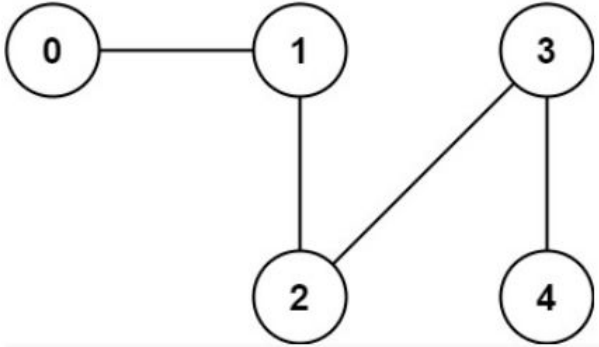
- Given an unweighted **undirected** graph, count the number of connected components (CC)
  - CC is a subgraph, and every node in the subgraph can reach all others in the same subgraph
- int countComponents(int n, vector<vector<int>>& edges)
  - n = is the number of nodes
  - edges: edge list where edges[i] is a vector of 2 numbers [a_i, b_i]
  - No multiple edges or self loops

Example 1:




```
Input: n = 5, edges = [[0,1],[1,2],[3,4]]
Output: 2
```

Example 2:




```
Input: n = 5, edges = [[0,1],[1,2],[2,3],[3,4]]
Output: 1
```

"Acquire knowledge and impart it to the people."

"Seek knowledge from the Cradle to the Grave."