Problem 1743: Restore the Array From Adjacent Pairs

Problem Information

Difficulty: Medium
Acceptance Rate: 74.94%
Paid Only: No
Tags: Array, Hash Table, Depth-First Search

Problem Description

There is an integer array `nums` that consists of `n` **unique** elements, but you have forgotten it. However, you do remember every pair of adjacent elements in `nums`.

You are given a 2D integer array `adjacentPairs` of size `n - 1` where each `adjacentPairs[i] = [ui, vi]` indicates that the elements `ui` and `vi` are adjacent in `nums`.

It is guaranteed that every adjacent pair of elements `nums[i]` and `nums[i+1]` will exist in `adjacentPairs`, either as `[nums[i], nums[i+1]]` or `[nums[i+1], nums[i]]`. The pairs can appear **in any order**.

Return _the original array_ `nums` _. If there are multiple solutions, return**any of them**_.

Example 1:

Input: adjacentPairs = [[2,1],[3,4],[3,2]] **Output:** [1,2,3,4] **Explanation:** This array has all its adjacent pairs in adjacentPairs. Notice that adjacentPairs[i] may not be in left-to-right order.

Example 2:

Input: adjacentPairs = [[4,-2],[1,4],[-3,1]] **Output:** [-2,4,1,-3] **Explanation:** There can be negative numbers. Another solution is [-3,1,4,-2], which would also be accepted.

Example 3:

Input: adjacentPairs = [[100000,-100000]] **Output:** [100000,-100000]

Constraints:

* `nums.length == n` * `adjacentPairs.length == n - 1` * `adjacentPairs[i].length == 2` * `2 <= n <= 105` * `-105 <= nums[i], ui, vi <= 105` * There exists some `nums` that has `adjacentPairs` as its pairs.

Code Snippets

C++:

```
class Solution {
public:
vector<int> restoreArray(vector<vector<int>>& adjacentPairs) {

}
};
```

Java:

```
class Solution {
public int[] restoreArray(int[][] adjacentPairs) {

}
}
```

Python3:

```
class Solution:
def restoreArray(self, adjacentPairs: List[List[int]]) -> List[int]:
```