Problem 444: Sequence Reconstruction

Problem Information

Difficulty: Medium
Acceptance Rate: 29.94%
Paid Only: Yes
Tags: Array, Graph, Topological Sort

Problem Description

You are given an integer array `nums` of length `n` where `nums` is a permutation of the integers in the range `[1, n]`. You are also given a 2D integer array `sequences` where `sequences[i]` is a subsequence of `nums`.

Check if `nums` is the shortest possible and the only **supersequence**. The shortest **supersequence** is a sequence **with the shortest length** and has all `sequences[i]` as subsequences. There could be multiple valid **supersequences** for the given array `sequences`.

* For example, for `sequences = [[1,2],[1,3]]`, there are two shortest **supersequences** , `[1,2,3]` and `[1,3,2]`. * While for `sequences = [[1,2],[1,3],[1,2,3]]`, the only shortest **supersequence** possible is `[1,2,3]`. `[1,2,3,4]` is a possible supersequence but not the shortest.

Return `true` _if_ `nums` _is the only shortest_**supersequence** for _`sequences`_, or_`false` _otherwise_.

A **subsequence** is a sequence that can be derived from another sequence by deleting some or no elements without changing the order of the remaining elements.

Example 1:

Input: nums = [1,2,3], sequences = [[1,2],[1,3]] **Output:** false **Explanation:** There are two possible supersequences: [1,2,3] and [1,3,2]. The sequence [1,2] is a subsequence of both: [**_1_** ,**_2_** ,3] and [**_1_** ,3,**_2_**]. The sequence [1,3] is a subsequence of both: [**_1_** ,2,**_3_**] and [**_1_** ,**_3_** ,2]. Since nums is not the only shortest

supersequence, we return false.

Example 2:

Input: nums = [1,2,3], sequences = [[1,2]] **Output:** false **Explanation:** The shortest possible supersequence is [1,2]. The sequence [1,2] is a subsequence of it: [**_1_** ,**_2_**]. Since nums is not the shortest supersequence, we return false.

Example 3:

Input: nums = [1,2,3], sequences = [[1,2],[1,3],[2,3]] **Output:** true **Explanation:** The shortest possible supersequence is [1,2,3]. The sequence [1,2] is a subsequence of it: [**_1_** ,**_2_** ,3]. The sequence [1,3] is a subsequence of it: [**_1_** ,2,**_3_**]. The sequence [2,3] is a subsequence of it: [1,**_2_** ,**_3_**]. Since nums is the only shortest supersequence, we return true.

Constraints:

* `n == nums.length` * `1 <= n <= 104` * `nums` is a permutation of all the integers in the range `[1, n]`. * `1 <= sequences.length <= 104` * `1 <= sequences[i].length <= 104` * `1 <= sum(sequences[i].length) <= 105` * `1 <= sequences[i][j] <= n` * All the arrays of `sequences` are **unique**. * `sequences[i]` is a subsequence of `nums`.

Code Snippets

C++:

```
class Solution {
public:
bool sequenceReconstruction(vector<int>& nums, vector<vector<int>>&
sequences) {

}
};
```

Java:

```
class Solution {
public boolean sequenceReconstruction(int[] nums, List<List<Integer>>
sequences) {
```

```
        }
    }
}
```

Python3:

```
class Solution:
    def sequenceReconstruction(self, nums: List[int], sequences: List[List[int]])
    -> bool:
```