Previous (/articles/super-egg-drop/) Next (/articles/uncommon-words-from-two-sentences/)

886. Possible Bipartition (/problems/possible-bipartition/)

Aug. 11, 2018 | 44.1K views

Average Rating: 3.19 (37 votes)

Given a set of `N` people (numbered `1, 2, ..., N`), we would like to split everyone into two groups of **any** size.

Each person may dislike some other people, and they should not go into the same group.

Formally, if `dislikes[i] = [a, b]`, it means it is not allowed to put the people numbered `a` and `b` into the same group.

Return `true` if and only if it is possible to split everyone into two groups in this way.

Example 1:

```
Input: N = 4, dislikes = [[1,2],[1,3],[2,4]]
Output: true
Explanation: group1 [1,4], group2 [2,3]
```

Example 2:

```
Input: N = 3, dislikes = [[1,2],[1,3],[2,3]]
Output: false
```

Example 3:

```
Input: N = 5, dislikes = [[1,2],[2,3],[3,4],[4,5],[1,5]]
Output: false
```

Note:

1. `1 <= N <= 2000`
2. `0 <= dislikes.length <= 10000`
3. `1 <= dislikes[i][j] <= N`
4. `dislikes[i][0] < dislikes[i][1]`
5. There does not exist `i != j` for which `dislikes[i] == dislikes[j]`.

Solution

Approach 1: Depth-First Search

Intuition

It's natural to try to assign everyone to a group. Let's say people in the first group are red, and people in the second group are blue.

If the first person is red, anyone disliked by this person must be blue. Then, anyone disliked by a blue person is red, then anyone disliked by a red person is blue, and so on.

If at any point there is a conflict, the task is impossible, as every step logically follows from the first step. If there isn't a conflict, then the coloring was valid, so the answer would be `true`.

Algorithm

Consider the graph on `N` people formed by the given "dislike" edges. We want to check that each connected component of this graph is bipartite.

For each connected component, we can check whether it is bipartite by just trying to coloring it with two colors. How to do this is as follows: color any node red, then all of it's neighbors blue, then all of those neighbors red, and so on. If we ever color a red node blue (or a blue node red), then we've reached a conflict.

Java | Python

Copy

```python
class Solution(object):
    def possibleBipartition(self, N, dislikes):
        graph = collections.defaultdict(list)
        for u, v in dislikes:
            graph[u].append(v)
            graph[v].append(u)

        color = {}
        def dfs(node, c = 0):
            if node in color:
                return color[node] == c
            color[node] = c
            return all(dfs(nei, c ^ 1) for nei in graph[node])

        return all(dfs(node)
                   for node in range(1, N+1)
                   if node not in color)
```

Complexity Analysis

- Time Complexity: $O(N + E)$, where E is the length of `dislikes`.
- Space Complexity: $O(N + E)$.

Rate this article:

Previous (/articles/super-egg-drop/) Next (/articles/uncommon-words-from-two-sentences/)

Comments: 8 Sort By

Type comment here... (Markdown is supported)

Preview Post

(/calvinchankf)

calvinchankf (/calvinchankf) ★ 2656 May 29, 2019 3:12 AM

Similar logic. Here are the BFS as well as the recursive DFS approaches.

BTW, here is another question we can use the similar approach, #785 (https://leetcode.com/problems/is-graph-bipartite) , practice them in a row for better understanding 💪

Read More

14 ▲ ▼ Share Reply

(/jonas2018)

jonas2018 (/jonas2018) ★ 39 January 28, 2019 2:59 AM

I don't like the way the solution is styled with ambiguous variable names etc. So here's a refactored version

```
class Solution {
    List<List<Integer>> graph = new ArrayList<>();
```

Read More

12 ▲ ▼ Share Reply

SHOW 1 REPLY

(/user0320)

user0320 (/user0320) ★ 3 February 29, 2020 8:53 AM

DO CHECK double direction.

3 ▲ ▼ Share Reply

SHOW 1 REPLY

(/mangrer)

mangrer (/mangrer) ★ 91 May 13, 2020 9:15 AM

@awice, Looks like your solutions assume some good level understanding with graph concepts, at times, you have to assume that your audience is new to the topic (e.g., what bipartite means).

4 ▲ ▼ Share Reply

(/lefeizzz)

lefeizzz (/lefeizzz) ★ 117 August 14, 2018 2:10 AM

Elegant

1 ▲ ▼ Share Reply

(/zhangbaohe)

zhangbaohe (/zhangbaohe) ★ 31 July 28, 2019 3:42 AM

good

0 ▲ ▼ Share Reply

(/vincentfeng0101)

vincentFeng0101 (/vincentfeng0101) ★ 40 6 hours ago

can someone explain for me : if (!dfs(nei, c ^ 1)) why " c ^ 1 " ??

0 ▲ ▼ Share Reply

SHOW 2 REPLIES

(/kniffina)

kniffina (/kniffina) ★ 142 October 27, 2019 5:07 PM

@awice Why does a union find not work for this problem?

0 ^ v | Share | Reply

SHOW 1 REPLY