

785. 判断二分图

题目: Is Graph Bipartite?

语言: python3

英文版链接: <https://leetcode.com/problems/is-graph-bipartite/description/>

中文版链接: <https://leetcode-cn.com/problems/is-graph-bipartite/>

题目分析

对于本题来说, 如果可以用两种颜色对图中的节点进行着色, 并且保证相邻的节点颜色不同, 那么这个图就是二分图。

所以将本题转化成了一个着色问题, 同时用dfs进行遍历节点, 即可完成对所有的节点的涂色与判断。
需要注意的是我们用-1表示未着色, 0/1分别表示两种颜色。

答案

```
class Solution:
    def isBipartite(self, graph) -> bool:
        # 实际本题还是一个着色问题
        # 初始化一个节点颜色list -1表示未涂色, 0/1分别表示两种颜色

        colors = [-1] * len(graph)
        for i in range(len(graph)):
            if colors[i] == -1 and not self.dfs(i, 0, colors, graph):
                return False
        return True

    def dfs(self, cur_node, cur_color, colors, graph):
        if colors[cur_node] != -1:
            # 如果当前节点已涂色, 则直接判断两次颜色是否相同 并返回
            return colors[cur_node] == cur_color

        # 给节点涂色
        colors[cur_node] = cur_color
        for next_node in graph[cur_node]:
            # 1 - cur_color 表示与当前颜色相反的颜色
            # 深度遍历所有的节点
            if not self.dfs(next_node, 1 - cur_color, colors, graph):
                return False
        return True
```