

0207. 课程表

👤 [ITCharge](#) ⌚ 大约 2 分钟

- 标签：深度优先搜索、广度优先搜索、图、拓扑排序
- 难度：中等

题目链接

- [0207. 课程表 - 力扣](#)

题目大意

描述： 给定一个整数 $numCourses$ ，代表这学期必须选修的课程数量，课程编号为 $0 \sim numCourses - 1$ 。再给定一个数组 $prerequisites$ 表示先修课程关系，其中 $prerequisites[i] = [ai, bi]$ 表示如果要学习课程 ai 则必须要先完成课程 bi 。

要求： 判断是否可能完成所有课程的学习。如果可以，返回 `True`，否则，返回 `False`。

说明：

- $1 \leq numCourses \leq 10^5$ 。
- $0 \leq prerequisites.length \leq 5000$ 。
- $prerequisites[i].length == 2$ 。
- $0 \leq ai, bi < numCourses$ 。
- $prerequisites[i]$ 中所有课程对互不相同。

示例：

- 示例 1:

输入: `numCourses = 2, prerequisites = [[1,0]]`

输出: `true`

解释: 总共有 2 门课程。学习课程 1 之前，你需要完成课程 0。这是可能的。

py

- 示例 2:

输入: numCourses = 2, prerequisites = [[1,0],[0,1]]

输出: false

解释: 总共有 2 门课程。学习课程 1 之前, 你需要先完成课程 0; 并且学习课程 0 之前, 你还应先完成课程 1。这是不可能的。

解题思路

思路 1: 拓扑排序

1. 使用哈希表 *graph* 存放课程关系图, 并统计每门课程节点的入度, 存入入度列表 *indegrees*。
2. 借助队列 *S*, 将所有入度为 0 的节点入队。
3. 从队列中选择一个节点 *u*, 并令课程数减 1。
4. 从图中删除该顶点 *u*, 并且删除从该顶点出发的有向边 $\langle u, v \rangle$ (也就是把该顶点可达的顶点入度都减 1)。如果删除该边后顶点 *v* 的入度变为 0, 则将其加入队列 *S* 中。
5. 重复上述步骤 3 ~ 4, 直到队列中没有节点。
6. 最后判断剩余课程数是否为 0, 如果为 0, 则返回 True, 否则, 返回 False。

思路 1: 代码

```
import collections

class Solution:
    def topologicalSorting(self, numCourses, graph):
        indegrees = {u: 0 for u in graph}
        for u in graph:
            for v in graph[u]:
                indegrees[v] += 1

        S = collections.deque([u for u in indegrees if indegrees[u] == 0])

        while S:
            u = S.pop()
            numCourses -= 1
            for v in graph[u]:
                indegrees[v] -= 1
                if indegrees[v] == 0:
                    S.append(v)
```

```
    if numCourses == 0:
        return True
    return False

def canFinish(self, numCourses: int, prerequisites: List[List[int]]) ->
bool:
    graph = dict()
    for i in range(numCourses):
        graph[i] = []

    for v, u in prerequisites:
        graph[u].append(v)

    return self.topologicalSorting(numCourses, graph)
```

思路 1：复杂度分析

- **时间复杂度：** $O(n + m)$ ，其中 n 为课程数， m 为先修课程的要求数。
- **空间复杂度：** $O(n + m)$ 。