Description Solution Discuss (999+) Submissions

C#

210. Course Schedule II

Medium 5856 219 Add to List Share

There are a total of `numCourses` courses you have to take, labeled from `0` to `numCourses - 1`. You are given an array `prerequisites` where `prerequisites[i] = [a_i, b_i]` indicates that you **must** take course `b_i` first if you want to take course `a_i`.

- For example, the pair `[0, 1]`, indicates that to take course `0` you have to first take course `1`.

Return *the ordering of courses you should take to finish all courses*. If there are many valid answers, return **any** of them. If it is impossible to finish all courses, return **an empty array**.

Example 1:

```
Input: numCourses = 2, prerequisites = [[1,0]]
Output: [0,1]
Explanation: There are a total of 2 courses to take. To take course 1 you should have finished course 0. So the correct course order is [0,1].
```

Example 2:

```
Input: numCourses = 4, prerequisites = [[1,0],[2,0],[3,1],[3,2]]
Output: [0,2,1,3]
Explanation: There are a total of 4 courses to take. To take course 3 you should have finished both courses 1 and 2. Both courses 1 and 2 should be taken after you finished course 0.
So one correct course order is [0,1,2,3]. Another correct ordering is [0,2,1,3].
```

Example 3:

```
Input: numCourses = 1, prerequisites = []
Output: [0]
```

```
7               // Create t
adjacency list represen
the graph
8               for (var j
prerequisites.Length; j
9               {
10                  int des
prerequisites[j][0];
11                  int src
prerequisites[j][1];
12
 if(adjList.ContainsKey
13                  {
14
 adjList[src].Add(dest)
15                  }
16                  else
17                  {
18                      Lis
lst = new List<int>();
19
 lst.Add(dest);
20
 adjList.Add(src, lst);
21                  }
22
23                  // Reco
degree of each vertex
24                  indegre
+= 1;
25              }
26
27              // Add all
with 0 in-degree to the
28              Queue<int>
Queue<int>();
29              for (int i
```

Testcase Run Code Result

Accepted Runtime: 277 ms

Your input
2
[[1,0]]

Output [0,1]

Expected [0,1]

Console Use Example Testcase