Course Schedule

Try to solve the Course Schedule problem.

We'll cover the following

- Statement
- Examples
- Understand the problem
- Figure it out!
- Try it yourself

Statement

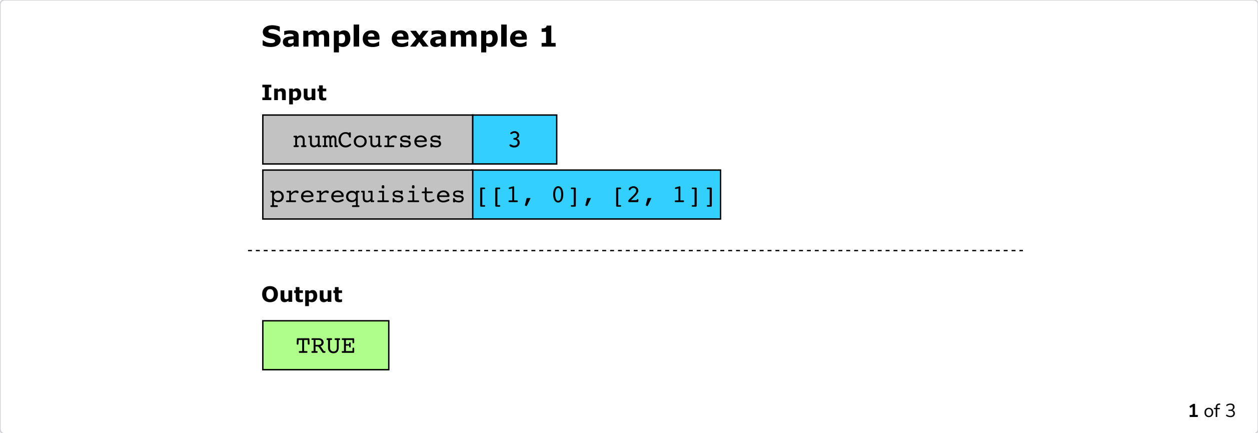
There are a total of `numCourses` courses you have to take. The courses are labeled from `0` to `numCourses - 1`. You are also given a `prerequisites` array, where `prerequisites[i] = [a[i], b[i]]` indicates that you must take course `b[i]` first if you want to take the course `a[i]`. For example, the pair $[1, 0]$ indicates that to take course 1, you have to first take course 0.

Return TRUE if all of the courses can be finished. Otherwise, return FALSE.

Constraints:

- $1 \leq$ `numCourses` ≤ 2000
- $0 \leq$ `prerequisites.length` ≤ 5000
- `prerequisites[i].length` $= 2$
- $0 \leq$ `a[i]`, `b[i]` $<$ `num_courses`
- All the pairs `prerequisites[i]` are unique.

Examples



Sample example 1

Input

numCourses	3
prerequisites	[[1, 0], [2, 1]]

Output

TRUE

1 of 3

Understand the problem

Let’s take a moment to make sure you've correctly understood the problem. The quiz below helps you check if you're solving the correct problem:

Course Schedule

1 What is the output if the following prerequisites and number of courses are given as input?

prerequisites = [[1, 0], [1, 2], [3, 1], [4, 1], [1, 4], [5, 1]]

numCourses = 6

A) TRUE

B) FALSE

Submit Answer

Question 1 of 3
0 attempted

Reset Quiz

> **Note:** The **in-degree** is the number of edges coming into a vertex in a directed graph.

Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.

Drag and drop the cards to rearrange them in the correct sequence.

Initialize a graph containing the key as the parent and the value as its child’s vertices.

Populate the in-degrees hash map.

Find all sources with 0 in-degrees.

For each source, append it to a sorted list, retrieve all of its children, and decrement the in-degrees of the respective child by 1.

Repeat the process until the source queue becomes empty.

Reset Show Solution Submit

Try it yourself

Implement your solution in the following coding playground.

```
import java.util.*;

class CourseSchedule {
    public static boolean canFinish(int numCourses, int[][] prerequisites) {
        // Write your code here

        return false;
    }
}
```

Powered by AI Submit

Test Cases Results

Case 1 Case 2 Case 3

Input #1

2

Input #2

[[1,0],[0,1]]

Course Schedule

← Back
Solution: Course Sche...

Next →
Solution: Course Sche...

☑ Mark as Completed