Graphs III

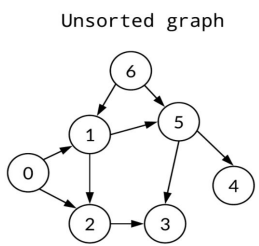
AGENDA & ANNOUNCEMENTS

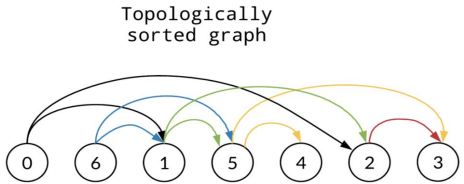
- Topological Sort
- Breakout Session
- Breakout Session Recap, Q&A
- *Ask questions in the Slack Channel*

Topological Sort

TOPOLOGICAL ORDERING

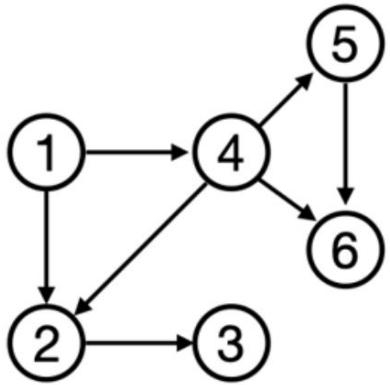
- “An ordering of nodes in a **directed acyclic graph** for each directed edge from node A to node B, node A appears before node B in the ordering”
- Only directed acyclic graphs (DAGs) have topological orderings
- A DAG can have multiple topological orderings
- In order to get the topological ordering, we use *topological sort*





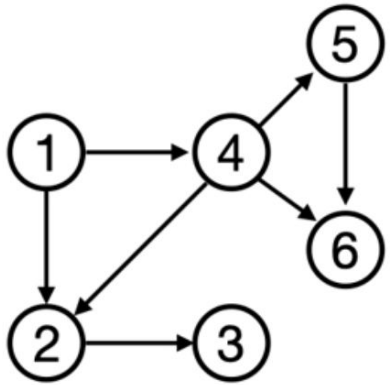
TOPOLOGICAL SORT

- Can be used to find the topological ordering of a DAG in O(V + E) time
- Algorithm
    1. Pick an unvisited node
    2. Do a DFS starting with that node and only explore unvisited nodes
    3. On the recursive callback of the DFS, add the current node to the topological ordering in reverse order
    4. Continue until all nodes are visited



TOPOLOGICAL SORT

- Topological sort is useful when figuring out an ordering of nodes that need to satisfy dependencies and requirements
- **Keywords**: prerequisites, requirements, dependencies



COURSE SCHEDULE II

- Leetcode Link

HOW TO SOLVE ANY GRAPH PROBLEM

1. Translate the problem into graph terminology

- What are the vertices, edges, weights (if needed)?

2. Build your graph

- Do you even need to build a graph? Should you use an adjacency matrix/list?

3. Traverse your graph

- Should you use BFS/DFS? Do you need an auxiliary data structure?

Breakout Session

With New People!

BREAKOUT SESSION

- Find whether it is possible to finish all tasks from given dependencies
- Find Eventual Safe States

Please make sure to do the mid-course survey!

Breakout Session Recap

BREAKOUT SESSION RECAP

- Find whether it is possible to finish all tasks from given dependencies
  - Remember the keywords for a lot of topological sorting questions (most of them sound very similar)
  - Find if there is a topological ordering for taking courses
- Find Eventual Safe States
  - Very similar to topological sorting, slight variant of DFS