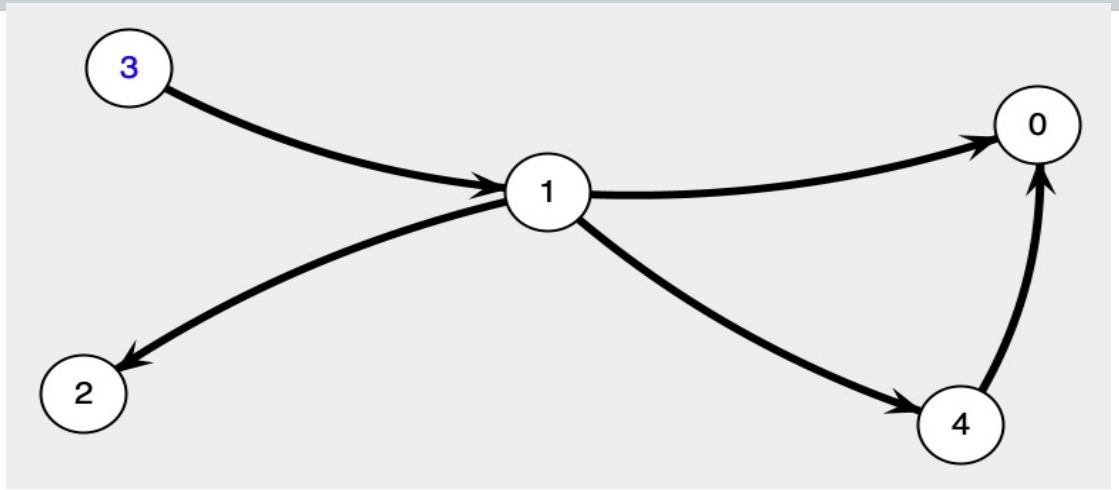
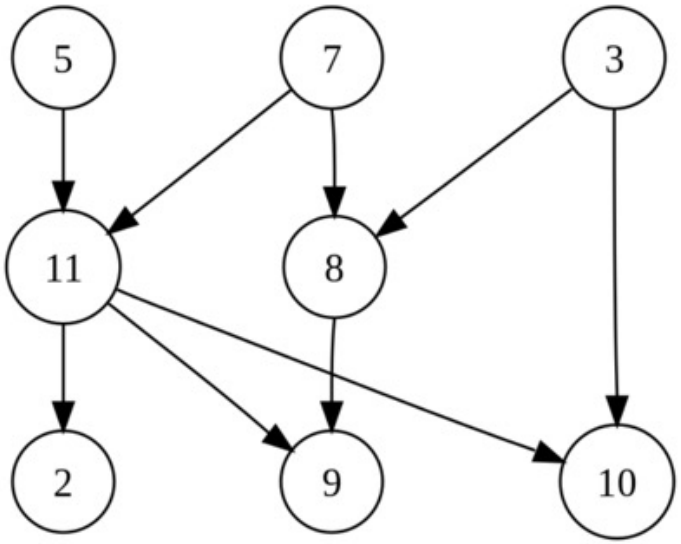
P4: Topological Sort



- Wikipedia: a **topological sort** or **topological ordering** of a directed graph is a linear ordering of its vertices such that for every directed edge$_{uv}$ from vertex *u* to vertex *v*, *u* comes before *v* in the ordering
  - **Valid:** (3, 1, 2, 4, 0) or (3, 1, 4, 0, 2)
  - **Invalid**: (1, 2, 4, 0, 3) or (3, 1, 0, 2, 4)

Topological Sort

- Ordering of vertices subjected to dependence defined by edges
  - If u→v, u is the predecessor and v is successor
  - All predecessors of a node v must come before v
- Applications in scheduling: instruction scheduling, class prerequisite checking, data serialization, project management, …



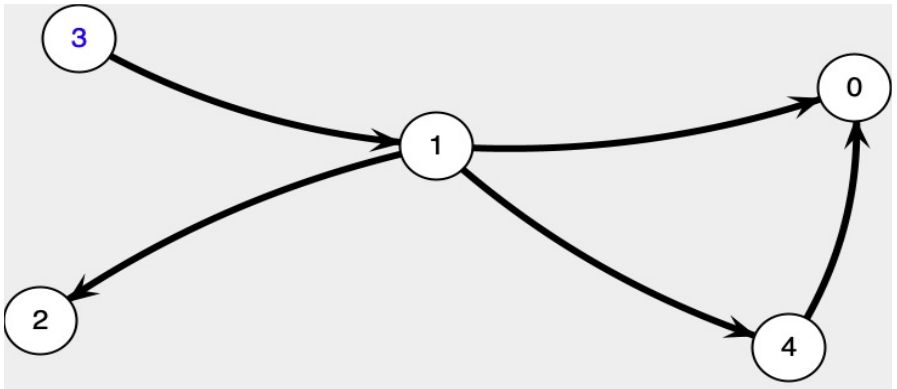
The graph shown to the left has many valid topological sorts, including:

- 5, 7, 3, 11, 8, 2, 9, 10 (visual top-to-bottom, left-to-right)
- 3, 5, 7, 8, 11, 2, 9, 10 (smallest-numbered available vertex first)
- 5, 7, 3, 8, 11, 10, 9, 2 (fewest edges first)
- 7, 5, 11, 3, 10, 8, 9, 2 (largest-numbered available vertex first)
- 5, 7, 11, 2, 3, 8, 9, 10 (attempting top-to-bottom, left-to-right)
- 3, 7, 8, 5, 11, 10, 2, 9 (arbitrary)

https://en.wikipedia.org/wiki/Topological_sorting

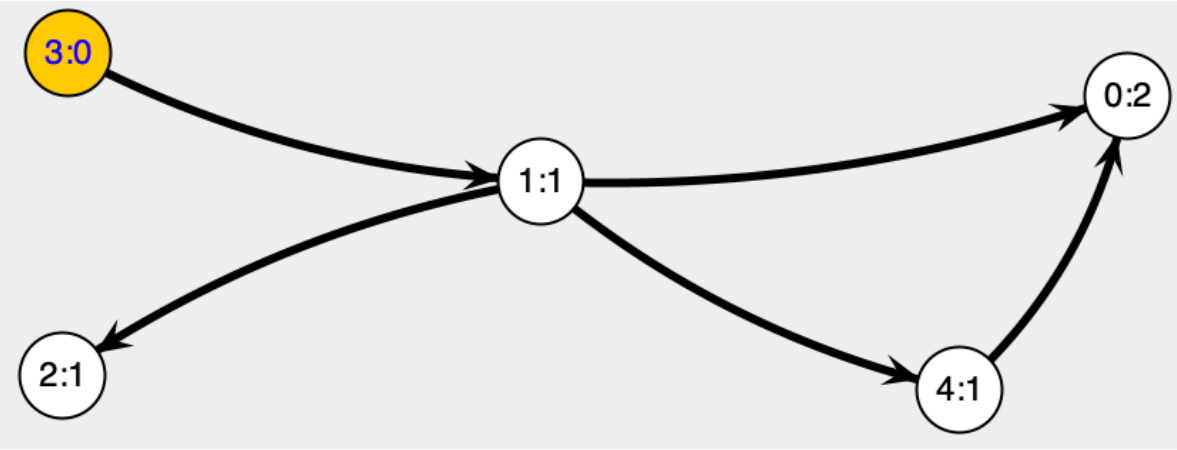
Kahn's Algorithm

- Idea: if a node has no incoming edges, it does not need to wait for other nodes to go first
  - Schedule nodes with a zero indegree first
  - Once a node is scheduled, its successor nodes might be able to get scheduled (i.e. wait for fewer predecessors)
  - If no nodes has a zero indegree: cycle in graph → topological sort impossible

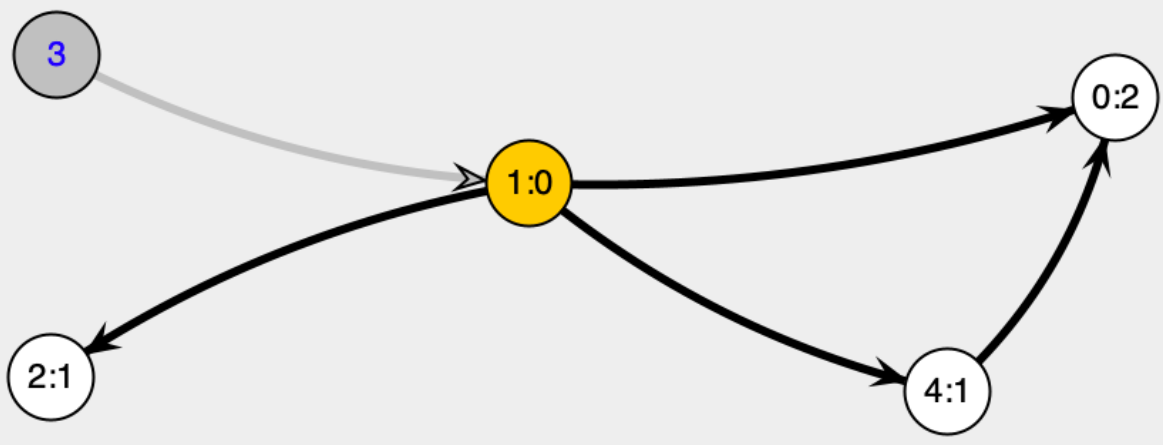


Which node to start?

Example

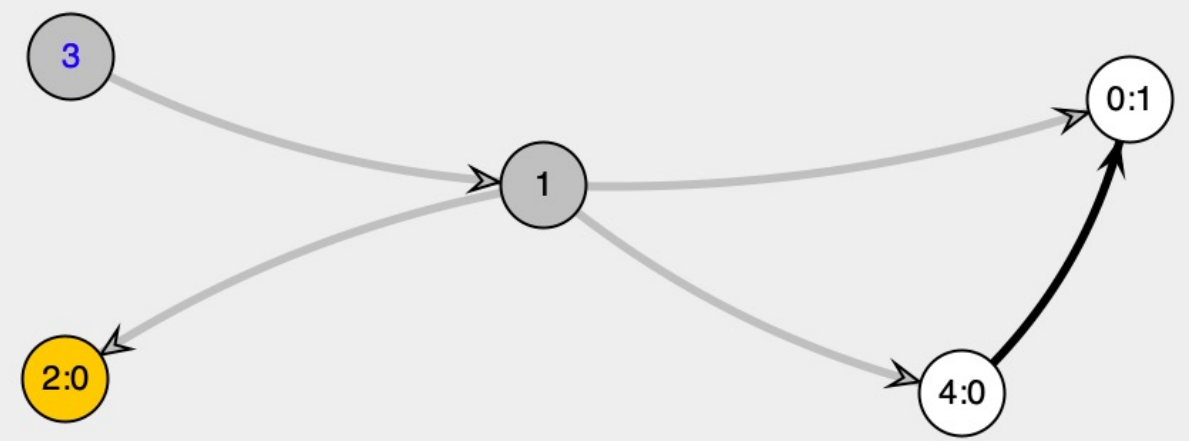


- Mark every node with its indegree (ID: cost)
- Select node with no incoming edges to start
- Sorted: 3

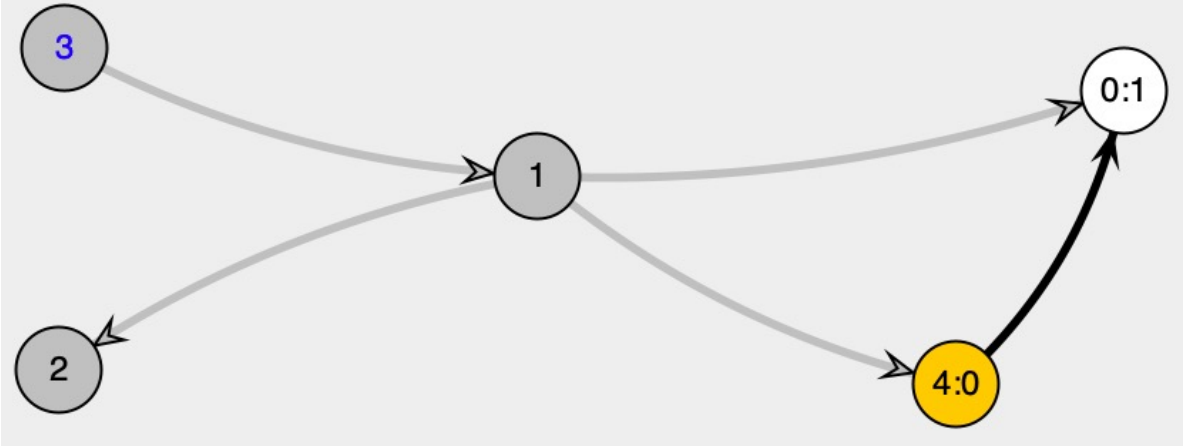


- Update successors of selected node
- Move to the next round
- Select node with no incoming edges
- Sorted: 3, 1

Example

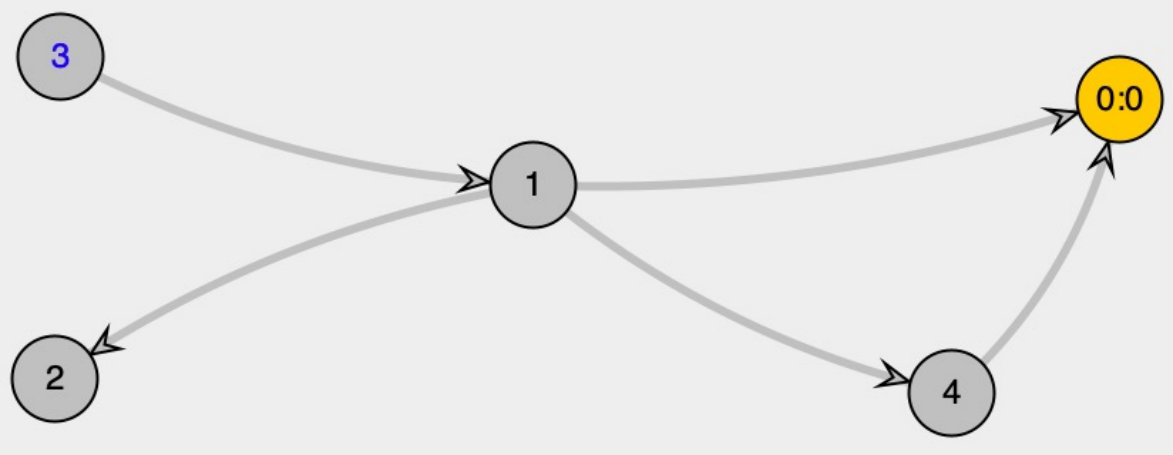


- Next round: updating and selection
- Pick lowest ID to break the tie
- Sorted: 3, 1, 2



- Sorted: 3, 1, 2, 4

Example



- Last node
- Sorted: 3, 1, 2, 4, 0