COMP20003
Algorithms and Data Structures
Topological Sort Algorithm

Nir Lipovetzky
Department of Computing and Information Systems
University of Melbourne
Semester 2

Directed Acyclic Graphs (DAGs)

Directed

Acyclic

Useful for modelling many problems:

- Temporal dependencies
- Causalities
- Hierarchies
- Compiling modularized programs

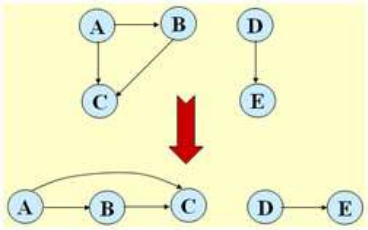
Topological sort

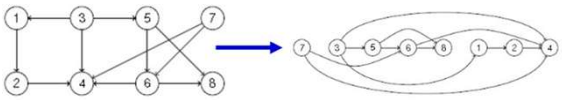
Topological sort: a partial ordering that fulfils certain constraints

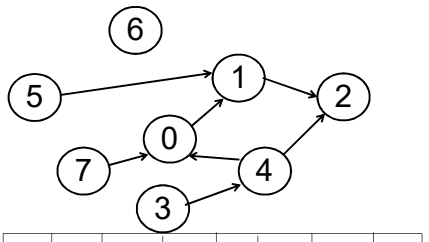
All edges e(i,j) go in horizontal i→j direction

The output can be the schedule for:

- A builder
- A course plan
- *etc.*







2	2	2	0	1	0	0	0	in-degree
0	1	2	3	4	5	6	7	

Topological sort construction: source removal algorithm

2	2	2	0	1	0	0	0	in-degree
0	1	2	3	4	5	6	7	

Identify a source (in-degree = 0)

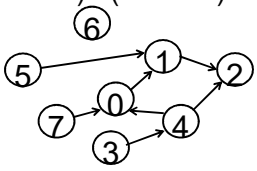
- put that node in the topsort output
- remove that node from DAG
- update in-degree matrix

Indentify another source…

Topological sort: sink removal

2	2	2	0	1	0	0	0	in-degree
0	1	2	3	4	5	6	7	

- 7-6-5-3-4(new src)-0(new src)-1(new src) - 2(new src)
- 6-3-4-7-0-5-1-2
- 5-7-3-4-0-1-2-6
- *etc.*



Topological sort: Assumptions

There must be at least one source and one sink for this topological sorting algorithm to work.

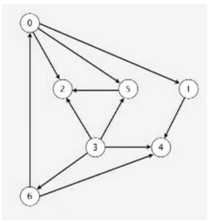
Is this a valid assumption?

Directed acyclic graph: must have at least one source and one sink

Topological sort: Assumptions

Complexity O()?

EXCERCISE



Topological sort: Uniqueness

If a Hamiltonian path exists in the DAG, then the topological sort is unique.

Finding a Hamiltonian Path is NP-Hard

How hard is to prove Uniqueness?

Have to solve the decision problem: Given a DAG, Does a Hamiltonian Path exists?

Topological sort: Uniqueness

The Hamiltonian Path problem has a property called polynomial verifiability

- Verifying the existence of a Hamiltonian path can be easier than determining its existence.

Given a topological sort, if two consecutive vertices are not connected, then you can swap them. Implies: Non Unique and No Ham. Path. Can be done in linear time.