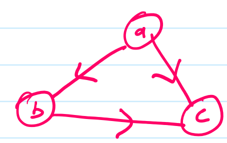
6.1 Topological Sort

Thursday, October 12, 2023 11:43 AM

- Sorting a digraph.

digraph.- directed graph.
a graph in which each edge has a direction.

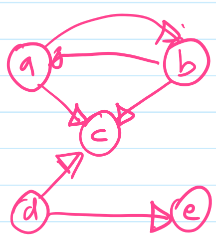
$G = \langle V, E \rangle$

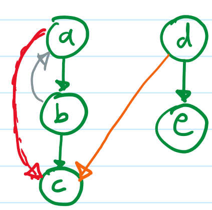


$V = \{a, b, c\}$

$E = \{(a,b), (b,c), (a,c)\}$

- DFS and BFS both work in digraphs





→ tree edges.
→ back edges.
→ forward edges.
→ cross edges.

- Digraphs can also be stored as
  - adjecency matrices
  - adjecency List.

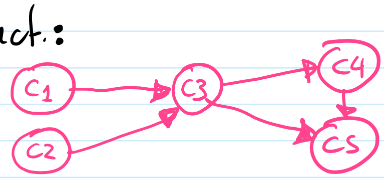
Motivating Example:

Suppose you have courses C1, C2, C3, C4, C5

- C1, C2 have no prerequisites
- C3 Requires C1 and C2
- C4 Req. C3
- C5 Req. C3 and C4

Kevin can only take one course per semester.
In which order should Kevin take the courses.

Abstract:



Problem (re-stated)

Problem (re-stated)

order the vertices $\langle v_0, v_1, v_2, v_3 \dots v_k \rangle$

in such ~~a~~ way that for every edge $\langle v_a, v_b \rangle$

$a < b$ in the order.

for every edge, the starting vertex appears before the ending vertex

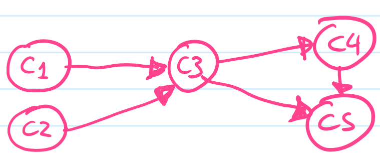
$\langle v_0, v_1, v_2, v_3 \dots v_k \rangle$

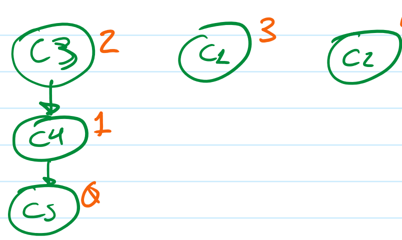
Note: only possible on digraphs without cycles.

DAGs: Directed Acyclic Graphs.

Solution #1: use DFS.

order the nodes by the order in which they become dead ends.



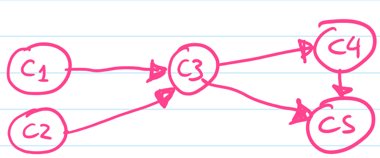


$\langle C5, C4, C3, C1, C2 \rangle$

reverse it:

$\langle C2, C1, C3, C4, C5 \rangle$

Solution #2: Decrease & conquer



$G = \langle V, E \rangle$ $\quad V = \{C_1, C_2, C_3, C_4, C_5\}$

$E = \{ \langle C_1, C_3 \rangle \langle C_3, C_4 \rangle \langle C_4, C_5 \rangle \langle C_2, C_3 \rangle \langle C_3, C_5 \rangle$

size = 5, how can I exploit somebody that can solve size = 4?

(i) Get rid of one of the nodes.

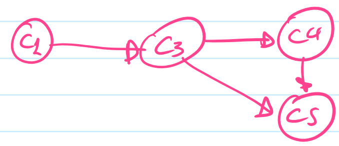
"Source" nodes, nodes with no incoming edges

Algorithm. C2

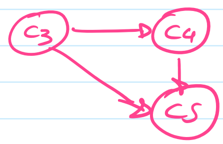
Algorithm. C2

- Select a source node
- place it next in the order.
- Simplify the graph.

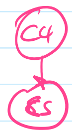
$\langle C_2$



$\langle C_2, C_1$



$\langle C_2, C_1, C_3,$

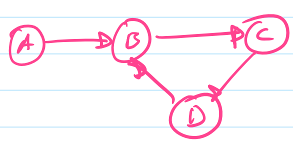


$\langle C_2, C_1, C_3, C_4$

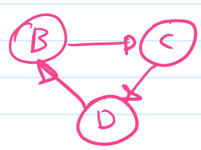
$\langle C_2, C_1, C_3, C_4, C_5 \rangle$

- What if the digraph has cycles?
  the algorithm will fail.

e.g.



$\langle A,$



Fail!!

EOF.