DAG: directed acyclic graph

no (directed) cycles

not a DAG
Topological sort (on a DAG)

- DFS tree from $V_1$
- "flattened" & pointing $L \rightarrow R$
- "Sort" all vertices (place in line)
  s.t. all directed edges are $\rightarrow$

No info about $V_2$ vs $V_3$
Topological sort (on a DAG)

- DFS tree from $V_1$

  "flattened" & pointing $L \rightarrow R$

  "Sort" all vertices (place in line) s.t. all directed edges are $\rightarrow$

  $V_4 \rightarrow V_1 \rightarrow V_3 \rightarrow V_2 \rightarrow V_5$

  implied

- No info about $V_2$ vs $V_3$
We need this order:

\[ V_1 \rightarrow V_3 \rightarrow V_2 \]

no info about \( v_2 \) vs \( v_3 \)

tree from \( v_1 \)

notice, we visited \( v_2 \) before \( v_3 \)

Otherwise
Rule: sort/output by finish time. v₂ finishes first. Then v₃. Then v₁.
OR

\begin{align*}
&V_1 \rightarrow V_3 \rightarrow V_2 \quad \text{group 1} \\
&V_4 \rightarrow V_5 \quad \text{group 2}
\end{align*}

\text{Continue DFS: search } V_4, V_5

\text{group 1 finished before group 2, so:}

\text{We could have had other groups or DFS trees, but each would give a valid topological sort}

\text{can add these; } V_4 \text{ found } V_3, V_1, \text{ but they were marked}
Intuition

DFS tree from v.

If there is an x→y relation
not in tree,
then x was explored
after y.

So y finished first
& we output correctly
Intuition

DFS tree from $v$.

$x \rightarrow y$ is implied in the tree

$y$ explored after $x$

So $y$ finished first & we output correctly
Intuition

(possible)

DFS tree from v.
Intuition

DFS tree from v.

Conclusion: if \( x \sim \sim \sim y \) then y finishes before x.

similar... if search started at v then it will be done before it gets discovered
Run DFS in any order.

When a vertex $v$ has been processed entirely, add it to a list.

Every vertex reachable from $v$ will be done before $v$, so it will already be in the list.

Similarly, $v$ will be ahead of any vertex that can reach it.