Applications of DFS

Topological Sort

Given dir graph $G = (V, E)$,
find a vertex ordering s.t.
$\forall (u, v) \in E \Rightarrow u$ appears before $v$

Given the directed acyclic graph (DAG):

```
a -> b -> c -> d -> e
```

$a \ c \ b \ e \ d$

Remark: if $G$ has cycle, no solution exists
So assume $G$ is acyclic (DAG)

First idea - find a source vertex $u$
- $\text{in-deg}(u) = 0$

Output $u$
Remove $u$ & repeat

How to find a source?
- Run DFS All($G$)
- Pick $u$ = last vertex finished

Example:

```
a -> b -> c -> d -> e
```

```
a -> f -> g -> h -> i
```

```
a -> b -> c -> d -> e
```

```
a -> b -> c -> d
```

```
a -> b -> c -> d
```

```
a -> b -> c
```

```
a -> b
```

```
```
DFS trees

Correctness Pf (Sketch): By contradiction.

Final Alg:
1. run DFSALL(G)
2. Output vertices in reverse order of finish

⇒ \(O(m+n)\) time

Corollary
\(\exists\) topological sort \(\iff G\) is a DAG.

Strongly Connected Components
Given digraph $G = (V, E)$, partition $V$ into components s.t.

- $u, v$ in same component
  - $\iff$ $\exists$ path $u \leadsto v$
    - and $\exists$ path $v \leadsto u$

**Example:**

- $\{i, j, e\}$, $\{a, b, c, d\}$, $\{g, h\}$, $\{f\}$, $\{k\}$

  (appl'n - control flow in programs, ...)
  (Simplifying digraphs into DAG)

- naive approaches
  - test reachability for every pair
    - by $O(n^2)$ BFS/DFS $\Rightarrow O(n(m + n))$ time
  - find cycles
    - contract, repeat

**History:** Purdom '68 $O(n^2)$ $O(m + n \log n)$
History: Purdom '68 $O(n)$
Munro '71 $O(m+n\log n)$
Tarjan '72 $O(m+n)$ complicated
Kosaraju '78 $O(m+n)$ simpler
Sharir '81

**first idea** - find a vertex $u$ in some source component of meta-graph
- find $u$'s component
- remove & repeat

**how to find in a vertex in a source component?**
- run DFSAll $(G)$
- pick $u =$ last vertex finished

**e.g.**
- DFS trees
- $a \rightarrow b \rightarrow c$
- $e \rightarrow d \rightarrow h \rightarrow g \rightarrow f$

Correctness Pf Sketch:
- all vertices $v$ s.t. $v \sim u$
- by running DFS in reverse graph $G^r$ from $u$
how to remove?
do nothing (no need to re-run the 2 DFSs)

Final Alg’m:
1. run DFSAll(G), label vertices by finish order
2. run DFSAll(G'), preferring larger labels when picking roots

Simple! $\implies \Theta(m+n)$ time

e.g. $G'$

a b c d e f $a$ $h$ $f$ $b$ $c$