

Depth-First Search and Topological Sort Review

DFS(G)

```
for each vertex  $v \in V$ 
  v.status = unvisited
  v.discover =  $\infty$ 
  v.finish =  $\infty$ 
  v.parent = NULL
time = 0
for each vertex  $v \in V$ 
  if v.status == unvisited
    DFS-visit(v)
```

DFS-visit(v)

```
v.status = open
time++
v.discover = time
for each neighbor u of v (edge (v,u) exists)
  if u.status == unvisited
    u.parent = v
    DFS-visit(u)
v.status = done
time++
v.finish = time
```

TopologicalSort(G)

```
run DFS(G), recording finish times for each vertex  $v \in V$ 
as vertex v is finished, add it to the front of a linked list of vertices
return the linked list of vertices (which is in reverse order of finish times)
```