Graphs
Undir: $m \leq \binom{n}{2}$, $\sum_v deg(v) = 2m$
Dir: $m \leq n * (n-1)$, $\sum_v indeg(v) = \sum_v outdeg(v) = m$
Subgraph: result of removing an edge, $V' \subseteq V, E' \subset E$
Induced SG: result of removing node. Is subgraph, $e \in E' \leftrightarrow (e \in E \wedge (u, v)) \in V$.
u connected to v: $(u \sim v) \rightarrow \exists$ path from u to v
G connected if $(u \sim v)$ $\forall (u, v)$

Representations
Adj. Matrix: row 1 = outgoing edge, col 1 = incoming edge
- Undir: $G = G^T$
- Find a neighbour: $O(n)$
- Access(v) = $O(1)$, traverse $V = O(n)$
- Traverse all edges of $v = O(n)$
- Traverse all edges of $G = O(n^2)$
- $O(V^2) = O(n^2)$ space

Adj. List: $\sum_v outdeg(v) = E$
- $O(V + E) = O(n + m)$ space
- Find a neighbour: $O(1)$
- Traverse all edges of $v = O(|neighbours(v)|)$
- Traverse all edges of $G = O(m)$
- Access(v) = $O(1)$, traverse $V = O(n)$

Trees
Tree: connected, acyclic graph
- Add edge $\rightarrow$ cycle; Remove edge $\rightarrow$ not connected
- n - 1 edges

Forest: graph with trees as connected components
Spanning tree: $T \subseteq G$ S.T $V(T) = V(G)$

BFS
- $O(n + m)$ time (list), $O(n^2)$ (matrix)
- $\Theta(n)$ space for both list and matrix
- Finds all nodes, finds shortest path from s to all others

BFS
- $O(n + m)$ time (list), $O(n^2)$ (matrix)
- Finds all nodes, finds shortest path from s to all others

Edge Classes
The edge (u, v) refers to the FOREST, not the graph.
- Tree: $(u, v) \in$ Forest
- Forward: v is descendant of u
- Back: v is ancestory of u (back = fwd in undirs)
- Cross: none of the above are true.
- Und: DFS tree/fwd only, BFS tree/cross only
- Dir: DFS: all edges, BFS: no fwd edges

DAGs
- Source: no incoming edges
- Sink: no outgoing edges
- Multiple sources/sinks are possible. At least one of each in every DAG.
- Source & Sink $\rightarrow$ no cycles
- A Digraph is a DAG $\leftrightarrow$ DFS has no back edges

Toposort
- Produces ordering s.t $(u, v) \in E \rightarrow u$ appears before v in the ordering
- Digraph has a Toposort $\leftrightarrow$ it is a DAG
- Run DFS, order in decreasing order of finish time
- $O(n + m)$
- Orders are not unique.

SCCs
- Run DFS. Run DFS on G^T, in decreasing order of ftime. The forests of the transpose DFS are the SCCs.
- Independent of toposort ordering