Directed Graphs

- DIGRAPH: graph with ONLY directed edges
  - Readability problem: Not nodes have an edge to connect them
  - Strongly connected: all nodes have edges between them

- Transitive closure of graph $G = G^*$
  - DIGRAPH with same verticies as G

- Floyd Warshall algo: Trans. closure
  - n steps: paths of length 1 to n, no more
  - $O(n^3)$: doesn't depend on number of edges
    - Still really slow

Give you an example

  - if $(v_i, v_k) \in G_{k-1} \wedge (v_k, v_j) \in G_{k-1} \wedge (v_i, v_j) \notin G_k$
    add new connection between

- BFS, DFS: work the same way but we need to it according to DIRECTIONS

- DDFS: edges = (discovery (tree) V non-tree) types
  - Discovery edges form a tree rooted at starting vertex
    - No loops
  - Non tree:
    - Backtrack
    - Forward
    - Cross

- DDFS: Garbage collection
  - memory deallocation: mark and sweep
    - (!) Assign (live ∨ ¬live) mark to object #useDFS

(!) EXAM: Floyd Warshall, DDFS

- Eulerian trail: path in finite graph where each edge visited exactly once
  - Pen has to leave the paper when drawing edges: NOT Eulerian!
  - nodes MUST have even degree of edges (think about it...)