

## Summary of topics discussed this week – 11/04/14

1. Post order
  - a. traverse the graph using DFS and list the vertices in postorder
  - b. note: first vertex in post order is a sink (no outgoing edges)
2. Reverse post order
  - a. find the post order of a graph using DFS and state them in reverse order
  - b. Note: last vertex in reverse post order is a sink
3. topological sort
  - a. find the reverse post order of a graph using DFS
  - b. this is a topological order where edges only go forward
4. BFS on a digraph
  - a. Traverse using a queue. Mark each node with a number starting with 0
5. connected components
  - a. ugraphs – two vertices are connected
6. strong components
  - a. digraph –  $\text{connected}(v,w)$  &  $\text{connected}(w,v)$
7. Finding SC's in a graph
  - a. compute the reverse post order of  $G^r$
  - b. run the DFS in the order of  $G^r$
8. MST
  - a. cut – a separation of vertices into two disjoint sets
  - b. crossing edge – connects a vertex in one set to the vertex in the other set
  - c. cut property – given any edge, the crossing edge of min weight is in MST
9. Two algorithms
  - a. Kruskals – sort edges by weight or maintain in a minPQ
    - i. keep adding edges to MST as long as an edge does not create a cycle
    - ii. need a minPQ and UF (of vertices) structures to implement this
  - b. Prim (lazy)
    - i. start with a specific vertex and add the minimum edge that is connected to it
    - ii. maintain a minPQ of edges
    - iii. no need to remove edges that may not be useful from the minPQ (as they will be removed later)
  - c. Prim(eager)
    - i. maintain a minPQ of vertices : vertex    edge    weight
    - ii. adjust the edge and weight if there is a better way to get to that vertex using another edge
    - iii. need a minPQ with additional property of : decreasePriority
  - d. indexedPQ

- i. The big idea here is that, there is a new operation called `decreaseKey(index, key)` that allows updating an entry in a PQ
- ii. Maintain two structures
  - 1. a minPQ – each key has an index and
  - 2. indexed array – that tells which index is a given key. So if a key needs to be changed, we can use the index to find the key (constant time) and decrease the key (log time, since we may have to swim up)