Dynamic trees (later: dynamic graphs)

6.897
L5.1
Feb. 15, 2005

Problem: maintain a forest of trees, including topology, where each node has unordered set of any # of children

Operations:

- **maketree:** return new vertex in new singleton tree
- **link(v, w):** make v a new child of w by adding edge (v, w)
  ↳ root of different tree than w's
- **cut(v):** delete edge (v, parent(v))
  ↳ not a root
- **findroot(v):** return root of tree containing v

Link-cut trees DS [Sleator & Tarjan – JCSS 1983]

- even though represented trees in forest may be unbalanced, can achieve $O(\lg n)$ time/operation (amortized)
- split represented tree into paths: (similar to Tango trees)
  - preferred child of node v
    = ith child if last access to v's subtree was in ith subtree
    none if last access to v's subtree was v or none.
  - preferred path = chain of preferred edges
  - preferred paths partition the nodes of the represented tree
- represent each preferred path by an auxiliary tree
  - splay tree keyed on depth
  - at root node, store path parent:
    path's top node's parent in represented tree
    (can't store path children – can be many at a node)
  - including these path-parent edges, we get "tree of auxiliary trees" – useful for analysis
    (can have large node degrees)