Tree ADT

- values stored in nodes
- each node links to some number of "children"
1. each node has EXACTLY one "parent"— except for one node, the "root"
2. a tree is connected: edges take us from the root to any node
3. a tree contains no cycle

Terminology:
- root: single node without a parent
- leaf: any node with no children
- internal nodes: node with children (with or without a parent)
- siblings: nodes with the same parent
- a "path" is a sequence of edges connected to each other
  (example: A–B–E is a path in the tree above)
- length of a path = number of edges on the path
  (example: A–B–E has length 2)
  Warning: some texts use number of _nodes_ instead
- "branching factor": maximum number of children of any node
  (example: 3 for the tree above)
- "depth" of a node = length of the path from root to that node
- "height" of a node = length of longest path from that node to a leaf

Tree traversal: going through every node in a tree to carry out a certain task with each node.
- "preorder": carry out the task with each node BEFORE carrying it out with each of the node's children
- "postorder": carry out the task with each of a node's children BEFORE carrying it out for the node itself