

BINARY SEARCH TREE: DELETE NODE

I followed these steps:

- 1) If the data to be deleted is in the tree THEN (2-5)
- 2) I get the parent node (that is, the node that references the data we want to delete).
- 3) If node is a leaf (has no children) THEN
 - a. I set the parent's reference to it as null. (effect: node deleted)
{eg parentNode.leftChild = null;}
- 4) If node has one child THEN
 - a. I set the parent to refer to the child of the node to be deleted
{e.g parentNode.leftChild = delNode.leftChild;}
- 5) If node has two children THEN (sub 1-2)
 1. I get the node's successor.
 - a. If node is root THEN
 1. I set the root to the successor obtained {root = successor} ELSE
 - b. If node to be deleted is not root THEN
 1. I set the parent to refer to the successor
{e.g parentNode.leftChild = successor;}
 2. I reconnect the detached tree by setting the new successor to refer to its appropriate child.
{e.g successor.leftChild = delNode.leftChild;}

The algorithm works because it carefully handles every case in the deletion process.