For the swapToLeaf with a single left child how does that work? Because obviously it can't be swapped with anything in the left subtree or the rules of a Binary Search Tree will be broken. Are we allowed to adjust the tree structure here so we have the same number of nodes and the node to be swapped just gets moved into position or must the tree maintain the same structure?

Thanks.

---

I guess what i'm asking is is this what should be done if we swap "quick" to a leaf:

```
0: "the"-2("the")-
1: "quick"("the")-l
2: "brown"("quick")-l
3: ="and"("brown")-l "fox"("brown")-r
4: ="dog"("fox")-l "jumped"("fox")-r
5: "girl"("jumped")-l "over"("jumped")-r
6: ="her"("girl")-r ="lazy"-2("over")-l
```

---

I made separate helper functions for finding the predecessor and successor.

to find the predecessor -> look at left child of the node and then go right until you find a leaf.
to find successor-> look at the right child of the node and then go left until you find a leaf

Subject: Re: Swap to Leaf
Posted by jherumin on Sun, 05 Mar 2017 20:36:18 GMT
View Forum Message <> Reply to Message

you swap for deletion, so any ordering issue with the value being swapped with its predecessor/successor will be fixed shortly after

Subject: Re: Swap to Leaf
Posted by adanderson8 on Sun, 05 Mar 2017 21:58:18 GMT
View Forum Message <> Reply to Message

So does that mean that leafToBST is just a helper function to pruneTree? So that even if the child is one you wouldn't just connect it's child to the parent but would swap and delete the leaf, always?

Subject: Re: Swap to Leaf
Posted by SSinischo on Sun, 05 Mar 2017 22:07:27 GMT
View Forum Message <> Reply to Message

adanderson8 wrote on Sun, 05 March 2017 15:58 So does that mean that leafToBST is just a helper function to pruneTree? So that even if the child is one you wouldn't just connect it's child to the parent but would swap and delete the leaf, always?

Subject: Re: Swap to Leaf
Posted by adanderson8 on Sun, 05 Mar 2017 22:10:43 GMT
View Forum Message <> Reply to Message

Ok, I finally got it. Went 3 hours down a rabbit hole and made some damn beautiful code to deal with what I thought but at least I'm on the right track now. Thanks for the help!!

Subject: Re: Swap to Leaf
Posted by adanderson8 on Sun, 05 Mar 2017 22:14:32 GMT
View Forum Message <> Reply to Message

One more quick question. If the frequency is above 1 then neither of these functions would determine that and decrease the frequency. That would be handled by a delete method written in
After finding the node in vbstDelete or rbtDelete, if the frequency is more than one you decrease the frequency. If not... Send it to the delete processes.

Ok thanks! I was just curious because there didn't seem to be a deleteBST tree method in the write up, it's there for rbt, just not for the BinarySearchTree. So i'll go ahead and implement that then!

The "delete" portion of the bst module is essentially swapNodeToLeaf and PruneBST. In delete vbst, you are essentially just calling swapNodeToLeaf then PruneBST.

Remember, due to my botching of the public interface for bst (missing a necessary callback in swapToLeaf), your deletion code in RBT will not be tested (it needs to be there, though).