

Csci 41: Introduction to Data Structures

Lab Exercise 9

Lab Instructor: Alex Liu

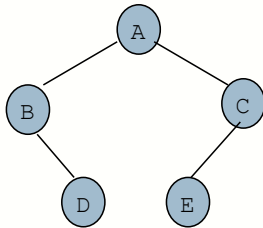
April 30, 2019

Objectives: Binary Tree and Binary Search Tree

Binary Tree: Use the description of the algorithms to convert them into source code.

Please try it without referencing to any material first. Only refer to material when really need

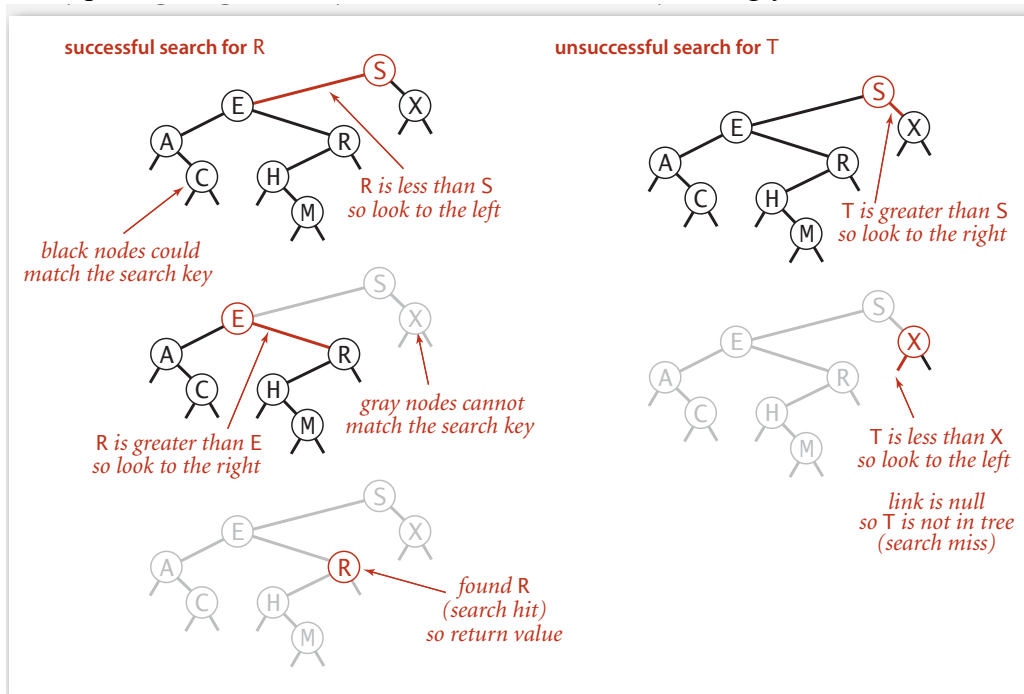
1. Implement a node class for Binary Tree (BT). Each node should record its *parent* node, left and right child nodes.
2. **Manually** construct a BT by following the below figure
3. Implement countLeaf function that computes the # of leaves in the BT.
4. Implement depth function that computes the depth of the BT.
5. Implement levelOrderScan that prints out all nodes' values in level order.
6. Implement lookup that finds whether a character passed in the lookup function can be found in the BT or not.
7. Implement size function that computes the number of nodes in the BT.
8. Implement minNode that returns the value (or Node* if you prefer) of the Node that has minimum value of the BT.



<<BST in the next page. This may help you prepare programming exam 3.>>

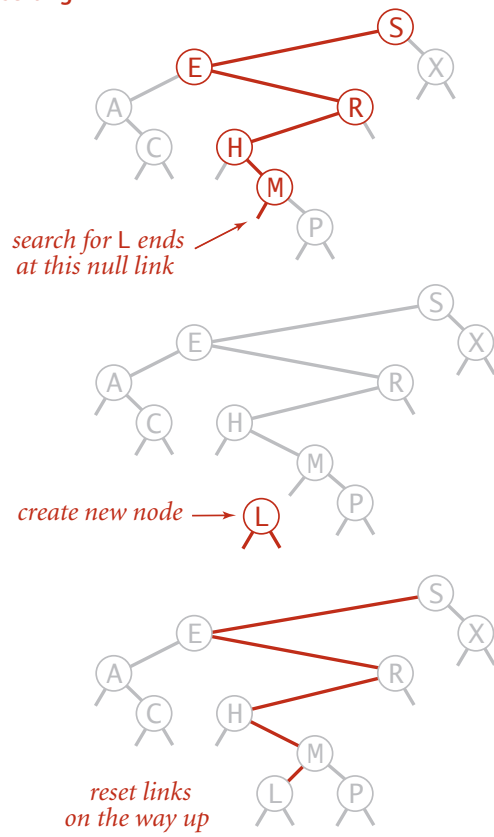
Binary Search Tree (BST): Use the description of the algorithms to convert them into source code. **Please try it without referencing to any material first. Only refer to material when really need**

1. The following diagram shows the algorithm of BST get/search function. Please implement a BST class and a search function accordingly.



2. The following diagram shows the algorithm of BST put/insert function. Please implement a BST class and an insert function accordingly.

inserting L



Insertion into a BST