sumRange Project

Robert Frenken

CSE 2331

This code creates a binary tree that will be used to insert a set of longs from a list to build the tree. It will then take the range of values from the 3rd Linux command and compute the sum of values found within the tree. The construction of the tree takes O(nh) time, where the insert function itself takes O(h) time. The computation to find the sum takes O(h) time. If there is no such triplet, the output will state that it couldn’t find a triplet.

Pseudocode:

dataX <- input from user

sumRange <- input from user

root <- new node

for element1 in dataX do:

root.Insert(element1)

for element1 in sumRange do:

sum <- root.range\_sum\_BST(root, i[0], i[1])

print('Range: ' + i + '. Sum: ' + sum)

class Node:  
  
 new Node(self, data):  
 data <- data  
 sum\_of\_children <- 0  
 left <- None  
 right <- None  
  
 insert(self, child):  
 if child < data:  
 if left is None:  
 left = Node(child)  
 sum\_of\_children += self.left.data  
 else:  
 sum\_of\_children += child  
 left.insert(child)  
 else:  
 if right is None:  
 right = Node(child)  
 sum\_of\_children += self.right.data  
 else:  
 sum\_of\_children += child  
 right.insert(child)  
  
 range\_sum\_BST(self, root, min, max):  
 sum = 0  
 if root is None:  
 return sum  
 if data > min:  
 sum += root.range\_sum\_BST(root.left, min, max)  
 if data < max:  
 sum += root.range\_sum\_BST(root.right, min, max)  
 if min <= data <= max:  
 sum += data  
 return sum