

Gebze Technical University
Computer Engineering

CSE 222
2017 Spring

HOMEWORK 5 REPORT

Deniz Can Erdem Yılmaz
151044001

3. Class Diagrams

Class diagrams can be found under the directory diagrams

5. Problem Solutions Approach

A binary tree class is the base of all homework 5. Binary tree class has a nested Node class to store data and hold reference of other nodes. Insertions intended to keep tree as a complete tree. No new level has reached until the current level has filled. Two constructors as no parameter and root parametered implemented to initialize the tree. Binary tree implements Iterable class to also implement iterator that traverses tree with pre-order. To supply pre-order traversal a stack used inside nested TreeIterator class. Iterator basically takes new node from stack, inserts right and left nodes in order to stack again and returns its current data starting from root. So all nodes return their data while going down of that node.

Binary search tree extends the binary tree class and only overrides the add method. It's because only difference between a binary tree and a binary search tree is the insertion of new element. If new element is lesser than the current add to left, else add to right is the motto of binary search tree. It again has 2 constructors as no parameter and root parametered to initialize the tree. Differently from binary tree, binary search tree also has levelOrderIterator class which can be predictable from its name that traverses the tree with level-order. To achieve this, queue structure used instead of stack. Iterator takes first element in queue, inserts left and right in order back to queue and returns current data. So every data returned as ordered levels.

Finally the family tree class extends binary tree class to have tree structure. But differently from binary tree, left of a node is its child and right of that node is its sibling. To achieve this two helper classes as ibn and ebu insertion methods implemented to insert each node correctly. Basically consider parsing three arguments as child, parent, nick. IbnInsertion finds name on nick, searches parent and inserts child if there is no child with same name. EbuInsertion finds parent, if child is same with name on nick, inserts it to parent's children. Else it searches name on the nick and inserts if search is successful. Same iterator as binaryTree (pre-order) used for test.