Binary Search Tree

Write a program in Java using Binary Search Tree data structure to manage information about persons (name: String which is **the key of the tree**, age: int)

1. void insert(String xName, int xAge) - check if the first letter of xName is 'B' (i.e. xName.charAt(0) == 'B') then do nothing, otherwise insert new person with name=xName, age=xAge to the tree.
2. Save all elements having age < the average age of the tree in format (name, age) to the file “q2.txt” by post-order traverse.
3. Calculate the height of the tree.
4. Calculate the number of nodes of the tree.
5. Delete the root of the tree by copying.
6. Perform breadth-first traverse from the root and delete by copying the second node having age >= the average age.
7. Check if the root having non-empty left-son then rotate it to right about its left-son.
8. Perform pre-order traverse from the root, rotate the third node having non-empty right-son then rotate it to left about its right-son and display the tree to the output screen.
9. Calculate balance factor of all nodes. Display all node with balance factor by breadth-first traverse.