**Question 2**

In this question you should complete some methods in **BSTree.java** files.

The class Car with 2 data members: owner and price is given and you do not need to edit it. The BSTree class is a binary search tree of Car objects. The variable **price is the key of the tree**. The following methods should be completed:

* void insert(string xOwner, int xPrice) - check if xOwner.charAt(0) = 'B' or xPrice>100 then **do nothing**, otherwise insert new car with owner=xOwner, price=xPrice to the tree.
* void **f1()** – You do not need to edit this function. Your task is to complete the insert(...) function above only. Output in the file **f1.txt** must be the following:

(A,5) (C,2) (E,4) (G,3) (D,6) (F,7)

(C,2) (G,3) (E,4) (A,5) (D,6) (F,7)

* void **f2()** – Perform pre-order traversal from the root but display to file f2.txt nodes having price in the interval [3,5] only. **Hint:** Copy the function preOrder(...) to preOrder2(...) and modify it. Output in the file **f2.txt** must be the following:

(C,6) (D,2) (F,4) (H,3) (I,5) (E,8) (G,7)

(F,4) (H,3) (I,5)

* void **f3()** – Perform breadth-first traversal from the root and delete by copying the first node having both 2 sons and price < 7. Output in the file **f3.txt** must be the following:

(C,8) (D,6) (E,9) (F,2) (G,7) (H,1) (I,3) (J,5) (K,4)

(C,8) (J,5) (E,9) (F,2) (G,7) (H,1) (I,3) (K,4)

* void **f4()** – Perform breadth-first traversal from the root and find the first node p having left son and price < 7. Rotate p to right about its’ left son. Output in the file **f4.txt** must be the following:

(C,8) (D,6) (E,9) (F,2) (G,7) (H,1) (I,3) (J,5) (K,4)

(C,8) (F,2) (E,9) (H,1) (D,6) (I,3) (G,7) (J,5) (K,4)