

1. Create a Huffman tree based on the character frequencies shown below.

<u>Character</u>	<u>Frequency</u>
Space	126
Newline	53
A	102
B	60
C	40
D	15
E	21
F	80
G	50

2. Implement the private method `leaves` in the `BinaryTree` class. Your implementation must be **recursive**. See the example on the accompany page.

```

public class BinaryTree {
//Implements a binary tree of strings
    private class Node {
        private Node left;
        private String data;
        private Node right;
        private Node(Node L, String d, Node R) {
            left = L;
            data = d;
            right = R;
        }
    }

    private Node root; //In a empty tree root is null

//ASSUME VALUES HAVE BEEN PUT IN THE TREE

    public String leaves() {
//if the tree is empty return the empty string otherwise
//return a string of the data in the leaves in the tree
        return leaves(root);
    }

    private String leaves(Node r) {
//if the r is null return the empty string
//otherwise return a string of the data in the leaves of
//the subtree rooted at r
    }
}

```

For the tree shown below the function leaves should return the string
MNO IJ UVW PQ

