

```

import java.util.Comparator;
import java.util.PriorityQueue;
import java.util.Scanner;

class HuffmanCoding {
    // Print Huffman codes
    public static void printCode(HuffmanNode root, String s) {
        if (root.left == null && root.right == null && Character.isLetter(root.c)) {
            System.out.println(root.c + ":" + s);
            return;
        }
        printCode(root.left, s + "0");
        printCode(root.right, s + "1");
    }

    // Main function
    public static void main(String[] args) {
        char[] charArray = { 'a', 'b', 'c', 'd', 'e', 'f' };
        int[] charfreq = { 5, 9, 12, 13, 16, 45 };
        PriorityQueue<HuffmanNode> q = new PriorityQueue<>(6, new MyComparator());

        for (int i = 0; i < charArray.length; i++) {
            HuffmanNode hn = new HuffmanNode();
            hn.c = charArray[i];
            hn.data = charfreq[i];
            hn.left = null;
            hn.right = null;
            q.add(hn);
        }

        HuffmanNode root = null;
    }
}

```

```

while (q.size() > 1) {
    HuffmanNode x = q.poll();
    HuffmanNode y = q.poll();
    HuffmanNode f = new HuffmanNode();
    f.data = x.data + y.data;
    f.c = '-';
    f.left = x;
    f.right = y;
    root = f;
    q.add(f);
}
printCode(root, "");
}
}

```

```

class HuffmanNode {
    int data;
    char c;
    HuffmanNode left;
    HuffmanNode right;
}

```

```

class MyComparator implements Comparator<HuffmanNode> {
    public int compare(HuffmanNode x, HuffmanNode y) {
        return x.data - y.data;
    }
}

```