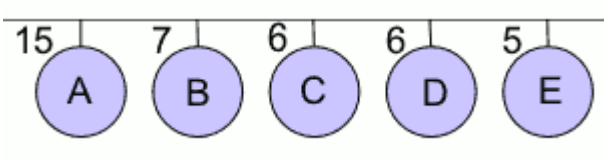
CS211 Lab 8

HUFFMAN CODING



Write a Java program that implements the Huffman coding algorithm.

Part I is to read in a piece of text from the user. The program should count how many times each letter appears in the text and print out that information. For example:

```
Enter your sentence: to be or not to be
' ' has a frequency of 5
'o' has a frequency of 4
't' has a frequency of 3
'b' has a frequency of 2
'e' has a frequency of 2
'r' has a frequency of 1
'n' has a frequency of 1
```

Part II is to create a new Binary Tree for each of the letters, add them into a Priority Queue and then keep combining the trees until only one is left (you can use the Tree and Node classes provided)

Part III (optional for 10/10) is to derive the Huffman encoding for the piece of text by using the resulting Huffman tree.

PEN AND PAPER EXERCISE

Create the Huffman tree for the following piece of text, derive the Huffman code and then calculate the rate of compression that has been achieved over 7-bit ASCII.

to be or not to be