

## Project 3

### Huffman Code Generator

Due: 7/16/13

Write a program that will read in a text file and generate a Huffman code based on its contents. Use the Huffman code generation algorithm discussed in class (also given in Chapter 15 of the textbook).

The input file will be named “message.txt” and will be placed in the same directory as your code. It might contain a message of any length, containing any characters (alphanumeric, punctuation, spaces, ...).

Your program should output to a text file “messageCode.txt” the codes generated for each unique character in the following format:

Frequency <sub>1</sub>	Code <sub>1</sub>	Character <sub>1</sub>
Frequency <sub>2</sub>	Code <sub>2</sub>	Character <sub>2</sub>
Frequency <sub>3</sub>	Code <sub>3</sub>	Character <sub>3</sub>
	...	
Frequency <sub>N</sub>	Code <sub>N</sub>	Character <sub>N</sub>

Figure 1 Output file (messageCode.txt) format

where **Character<sub>1</sub>** is the *most* frequently occurring character in the input file, Character<sub>2</sub> the second most frequently occurring character, and so on. Character<sub>N</sub> will be the character that appears the fewest number of times in the input file. Ties between characters which appear with equal frequency may be broken arbitrarily.

**Frequency<sub>i</sub>** = (the number of times Character<sub>i</sub> appears in the input file) / (total length of the input file). Clearly, this value will be between in the range (0,1].

**Code<sub>i</sub>** = the Huffman encoding for Character<sub>i</sub> generated by your program. In other words, a unique string of 0s and 1s that is not a subsequence of any other character’s code.