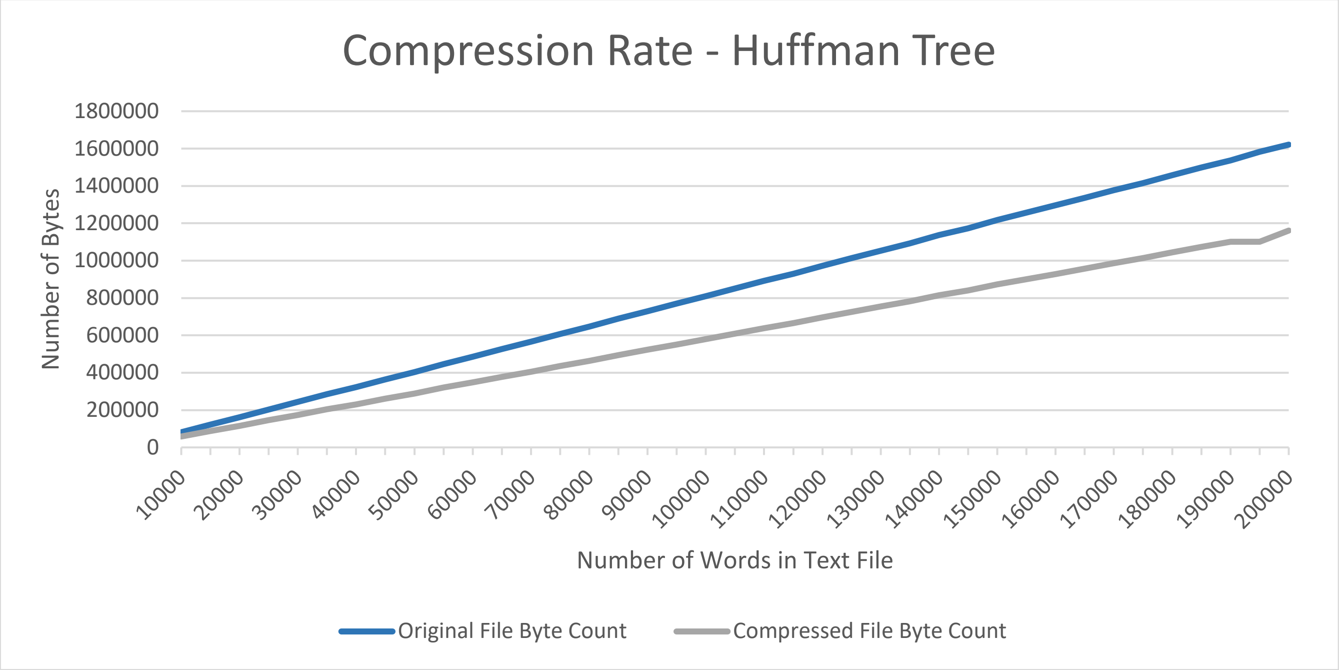
Kevin Claiborne

UID: u1080787

Assignment 12 Analysis

1. For my experiment, I set up an ArrayList in which I added a random String made up of characters between with lengths between 1 and 16. The ArrayList is then save as a text file with each word being added and a new line being added every 20 Strings. Once I have the text file I run the compression and record the number of bytes from the original and the number of bytes from the compressed file. I iterated the word count from 10,000 to 20,0000 stepping by 5,000 each time. The resulting plot is below.



2. Input files with little text and not so unique characters will result in little to no compression. While, larger files in which there is a lot of similar characters will result in much more compression. As you can see from the graph above the larger the file, the more it is compressed as the lines are not running completely parallel.

3. It seems to me that if one was to merge larger weight trees than the resulting tree would be lopsided. Because the number of elements with large amounts of occurrences would be considerably less than the number of elements with smaller amounts of occurrences. The tree would then be less effective.

4. Huffman’s algorithm performs lossless data compression because we are able to recover every bit and there is no missing information after decompression.

5. I spent about 10 hours on this assignment.