Michael Covarrubias PID#: A12409694
Mahaela Johnson PID#: A09635628

The output in the file that resulted from compressing checkpoint1.txt was a column of 256 numbers. All the numbers were 0 except for the numbers in positions representing a, b, c and d; each of the numbers in these positions had a values of 10. Which means a, b, c and d each had a frequency of 10. The last line was the compressed version of checkpoint1.txt was 1110010011100100111001001110010011100100111001001110010011100100111001001110

0100. The Huffman tree would look like
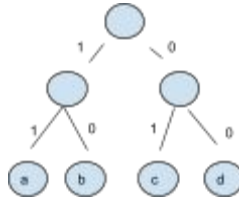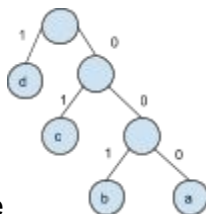
To build the tree, take that two symbols(nodes) with the smallest frequencies(if nodes have the same frequencies, take the nodes with the smaller ascii value) and make them the leafs of a tree. So a and b would be in a tree together. This tree can thought of as a new node with the a frequency equal to the sum of the leaf frequencies. Next, you have to take the next two nodes with the smallest frequency, c and d, and make them into a tree the same way a and b were. Now there are 2 trees, so you use a root node to connect them into a larger tree. To make a code for a symbol traverse the tree starting from the root and ending at the symbol. Each left traversal adds a 1 to the string of code, and each right traversal adds a 0.To find the so the encoding is a = 11, b =10, c = 01, d = 00. checkpoint1.txt contains "abcd" repeating 10 times. "abcd" encoded is 11100100. If you repeat 10 times, then the output is the same as the compressed output.

The output file that resulted from compressing checkpoint2.txt was a column of 256 numbers. All numbers were 0 except for lines representing a, b c and d. the numbers on those lines were 4, 8, 16, and 32 respectively. This means that a, b, c, and d had a frequency of 4, 8, 16 and 32 respectively. The last line was 000000001001001001010101010101010111111111111111111111111111111111110101010101010101001001001001000000

The Huffman tree would look like
Since a and b have the smallest frequencies, they can be paired to create leaves of a tree. Their combined frequency and the frequency of c are the next smallest frequencies, so c is added to the tree as a leaf with the same depth as the dummy node connecting a and b. Lastly a root node is placed to connect the existing tree and d. The encoding is a =000, b =001, c= 01 and d =1

So decoding the last line would result in

aabbbbcccccccccddddddddddddddddddddddddddddddddccccccccbbbbaa