I found this assignment to be very difficult to complete and the code walkthroughs during class helped a lot to complete it. I struggled a lot with writing the buildTreeFromMap function because writing recursive functions is still very tricky for me. I was able to complete it after much difficulty and a few rounds of testing. I still dont understand recursive functions. The only issue I wasnt able to resolve was adding line breaks. They would not be added to the map by the function although the space character was. I'm not really sure how I would resolve this, but some experimentation could yield a method to produce the line break character when a line break is detected. I think a future student could have an easier time completing this assignment by knowing how to solidly create and search through trees in whatever way the algorithm requires. I think the assignment was overall very fun. I got somewhat frustrated while working on the decodeBits function because I couldnt find a way to detect leaf nodes. I had just worked on the assignment for seven hours straight so I was not able to think staright. The next day I figured it out and the solution was way easier than what I first was trying. My advice is to take a lot more frequent breaks even if it feels "lazy". More frequent breaks could increase overall productivity. The most challenging part was definitely making the tree from the map. For me, creating a function recursively is the most difficult part to understand. I can definitely write them and eventually get them to work right but it always feels like guesswork. I think this is a very solid exercise in the Huffman algorithm. It provides a very solid foundation with building trees out of multiple types of class objects (huffmanNode, huffmanInternalNode, huffmanLeafNode). The only thing I would change is having a bit more time to turn in the asignment.