Design Diary 5,

A difficult task I encountered when working on this fifth milestone was that I was able to figure out , somewhat, use trie trees but figuring out how to use the search function correctly was giving me a hard time. Although I could use a preorder traversal to get the search function to work, I couldn't quite get my head to wrap around the idea of using it recursively because the root wasn't quite searching properly when I tried implementing it. However, I think I was able to make the addWord function work somewhat properly even though my search function was kind of wonky. Furthermore, since I couldn't quite grasp how to make the search function work, I wasn't able to correctly implement the auto-correct completion for the word that the user was typing and make the word pop up. In the end, I think I should have tried to implement the preorder traversal for the search function first and then tried to work my way through the addWord function because the addWord function had some correlation to the search function in which it added the word that wasn't in the trie tree and inputted it the tree if it didn't find it in the first place. An advice I could give to a future student when working on trie trees in general is to figure out/read first how trie trees work and start to figure out how to implement them on small subtrees, then go from there. In addition, read carefully and give yourself time to process the information before trying to implement too much code before knowing how it actually works.