

Course: CSC 340.04

Student: Melissa Estrada, SFSU ID: 918591991

Teammate: none

Assignment Number: 02

Assignment Due Date & Time: March-11-2019 at 11:59 PM

Part C - Dictionary 340 C++

Discussion #1:

At first I thought all I really had to do was convert my JAVA code into Cpp which turned out to be a lot harder then I thought. The analysis I took from the example output compared to the Java version was that it seemed fairly the same, but once I looked at the text file I realized I would have to do a lot more. So at first I realized I had to tackle loading the correct file into my program so that the correct data could be displayed. Making sure that the path to the file was correct was the first problem here I decided to have the user "search" for the file they wanted to load, which would be Data.CS.SFSU.txt from there I had to go to my next problem to solve which was having to split the words correctly so I could retrieve the correct definitions. I started off by simply writing the code to just display the contents of the file. Which I thought was a good way to get started that way I could see that it loaded properly, so I used fstream to open the file and close it so that the data could be sorted and stored. Then thought it would be a good idea to use the function strcmp so I could compare the characters but didn't work out for me. The way I loaded my data was with the cpp tokens that are the smaller individual units in a program which consists of keywords, identifies, constants, variables, and operators. By using vector as one of my data structures I was able to use functions such as begin() which returns an iterator pointing to the first element in the vector and end() which returns an iterator pointing to the theoretical element that follows the last element in the vector. I also used the size function that allowed me to return the number of elements in the vector. I also used the empty() function that returns whether the container is empty. I also used map because it is like a container that stores elements in a mapped fashion which allowed me to structure my program in a more efficient way but it was more confusing for some reason to get it to work since I couldn't figure out why I could pass the pair with three strings. I used the end(), size(), empty() and the pair insert(keyvalue, mapvalue) that added new elements to the map. I did want to incorporate the <regex> library since I did use it in my java program but it got to messy so I used the <iterator> library instead to make the program more efficient. With the algorithm library I used the find function that allowed me to find the values I needed within the range. One interesting thing I learned was the <cctype> (ctype.h) that declares a set of functions to classify and transform individual characters that I simply used to use tolower that converts uppercase letters to lowercase letters. Another useful class is the stringstream because object of this class use a string buffer that contains a sequence of characters the sequence accessed directly as a string object we use the str. That's how I was able to use the getline function as well as that get function as well. I felt like I choose to use these classes and functions because it was similar to my java program even though not everything translated over smoothly I felt like using a map and vector really helped me to get started but

there is a lot more time I need to invest in reviewing my data structures specially implementing them into cpp.

Discussion #2:

My program does not work properly. The first problem is that it will not display the definitions in alphabetical order I tried multiple different ways but I ended up changing me code so much that I ended up confused. What I tried to do was use the `sort()` that is part of `std::` but it just lead to more problems since I do have a two-dimensional array that I originally tried to make multi dimensional but that ended up not working out either. Since I was trying to make it possible for my third case to use the “distinct” but it just wouldn’t. It still works properly if I search just the word and the part of speech but it wont work if I include “distinct”. So after that didn’t work out I just let it stay at the 2 dimensional array and two cases since that was the only way it would compile. What I learned from this project was that the more search and googled what each library had to offer the more I came to a realization of the endless possibilities in what direction to head into I tried to be a little more ambitious and it ended up only confusing me. So I learned that sometimes the simple way probably might be the right way with time after this due date I do plan on trying to figure out where and how I went wrong to hopefully solving the problems I am left with. I am learning so much more efficiently making mistakes then ever before I am also coming to the realization that there are more ways to solve a problem and its ok to take on different approaches. So I will try to improve my program by reading more and figuring out how to implement the multi dimensional array.