John Sullivan

CSC212

Marco Alvarez

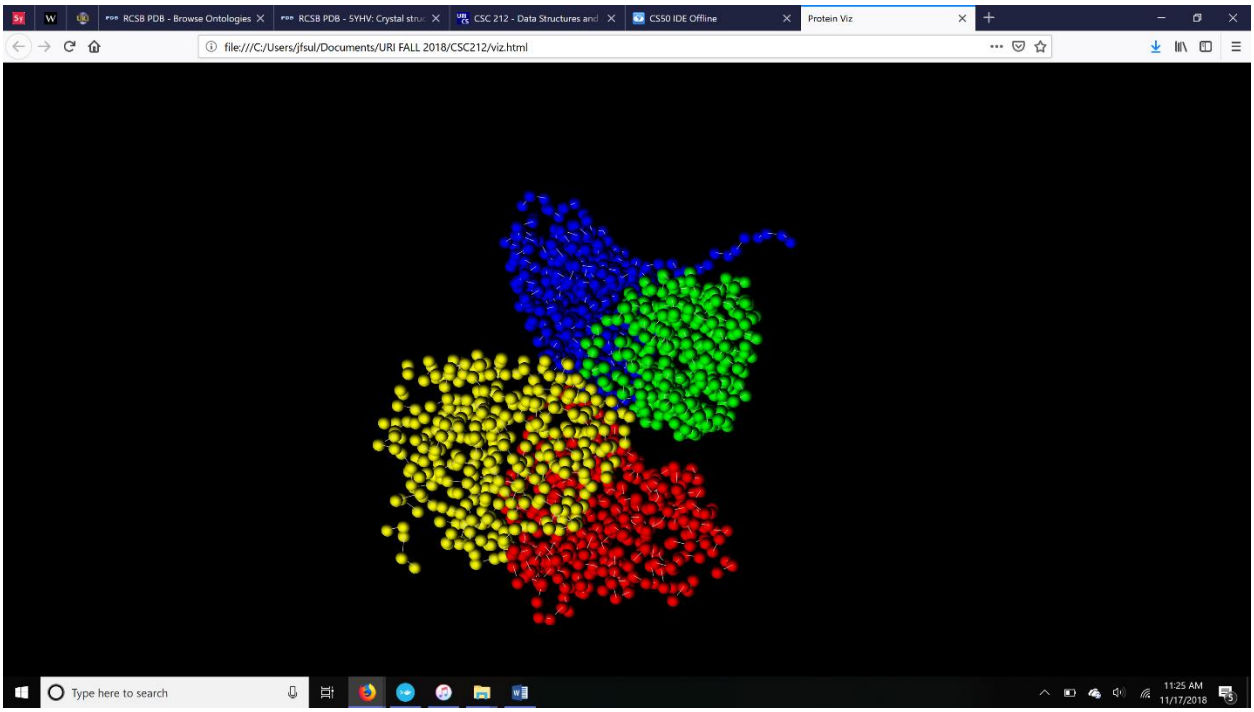
15 November 2018

Assignment #4 Report

1) The linked list I chose to use was just a single linked list. I thought of other options and I could have made it circular by simply having my last node point to the head rather than null, but I just wanted to implement my list without any further complication. My linked list has a few methods, one that calculates the Euclidian distance between points, one to calculate all of the nearest neighbors and store them in files, one to get the size of the list because I wasn't sure if I would need size at all, which I didn't end up needing, except for local testing purposes, an append function to create nodes in the list with given input and a print function which was also for local testing purposes.
2) My find neighbors method firstly opens two files, "coords.js" and "colors.js" before it does anything else. The reason I chose to do this within the method is because I didn't want to pass them as parameters, although I suppose I could have done it in the main and called the methods that way, but it works this way, so I'm not concerned. The method itself is called from main, and does not return anything, it just writes to the specified files like it's supposed to based on the specifications. I made two pointers, so I could check each node in the linked list and compare to every single other node using my function to calculate the Euclidian distance. After finding the nearest neighbor, I had a temporary pointer named "nearest" to hold the index of that node. Both nodes and their respective colors which are found with switch statements based on the chain ID are stored in two

files, “coords.js” and “colors.js” and once the function is done the program has run all the way through. The way I found the nearest neighbor was to calculate the Euclidian distance between each point which I had a separate method for in my class, so I would just call that distance function every time I wanted to compare two nodes using the respective x, y, z data for each node. The run time for my method is O(n^2) because it has two nested while loops checking every single node against each other. That is the most expensive part of the method.

3) **Crystal structure of an aminotransferase from Mycobacterium tuberculosis:**



⇨ **Continued on next page!**

An E. coli DPS protein from ferritin superfamily: