David Johnston

Professor Wei Kian Chen

CSI-281, Data Structures and Algorithms

2 September 2013

CSI-281 Programming Assignment 1 Report

I have decided to use a standard library list due to the difficulties that I imagine would arise when trying to remove an element from the middle of an array, vector, or deque. The list does of course have its downsides when it comes to frequent traversal, but I feel it’s worth it. I feel especially justified in light of the effects of removing small bits and pieces from the middle of a contiguous chunk of memory.

It seemed wise to me to contain the lists and the relevant functions within a class. This seemed advantageous in the general matters of composition and neatness, and did not really seem to have any downside.

The decision to have two lists – one for faculty and one for students – instead of one for everything was a difficult one to make. The primary advantage of the single list seemed to be some neat uses of polymorphism, including the one that the specifications of the assignment spell out so clearly in the display function. Ultimately though, I decided on two separate lists as it meant not having to worry about pointers (and the accompanying memory leaks), and having a much easier time writing all of the information back into text files.

The dual purpose (and long and silly name) of my search function is, I admit, rather dubious. I wanted my search function to do more than simply display the information of the person for whom I had searched. I tried to think of ways in which it might tell me the iterator position and list of the person (another downside to the separate lists I’m afraid, I could not do with just the iterator position) for whom I had searched. Unfortunately I could not figure out how I might get it to do such a thing, but I still wanted more from it than simple displaying of information. I think I found a semi-happy medium in returning the status of the person’s existence.