Lab 8

Author: Nigel Nelson
Course: CS2040
Date: 5/2/21

What did you learn from doing this lab?

- My biggest take away from this lab is how to use the list, vector, and map data structures from the standard template library in C++. Specifically, I learned about how to iterate over lists and vectors by using iterators. In addition, I learned how to manipulate these data structures with their included methods, which usually involved passing in an iterator as an argument. Passing iterators as arguments to data structure methods was a new concept to me so I learned how to correctly get the iterator to the desired position using the “advance” method. In addition, I learned how to delete objects contained in lists so that the heap could be deallocated of the space used by the created objects.

What did you find challenging about the lab?

- My biggest challenge for this lab was figuring out how to correctly navigate iterators to the correct position in lists and vectors so that I could use the standard template library methods to add elements, delete elements, etc. I initially attempted to use for loops, and after having issues with manipulating the data structures while iterating through them, I eventually was able to use these for loops to position my iterators successfully. However, I then found documentation on the “advance()” method for advancing iterators to a specified position, which ultimately made my code much simpler. In addition, I struggled with deallocating heap memory when using vectors and lists. I initially did not understand that the “erase()” and “clear()” methods did not also delete objects on the heap if the data structures contained pointers and not actual values. However, after realizing this I was able to understand that a combination of the “erase()” method and calling “delete” on the objects themselves correctly removed items from the dynamic data structures and also deallocated memory on the heap.

What would you recommend changing if this lab is reused in future years?

- There is not much that I would change for this lab in years going forward. I appreciated the provided code and provided UML diagram, as it lent a nice base so that I could focus more on learning how I could utilize the functionality provided by lists, vectors, and maps, as opposed to having to worry about having an object oriented design. Overall, the only thing that I would enjoy seeing in the future is perhaps a rubric so that students could have a better understanding of what key components of the lab to focus on.