**Purpose:**

This project was meant to test us on use of classes and different data types while reading from a file. Really it is the culmination of everything that we have worked on all semester.

**Procedure:**

We had to use a linked list, a heap, and a priority queue. The idea was to see if we still remember how to use it all but with a new challenge. Every single way uses the same readin function into a vector. That vector makes my life easy because I can get data out and put data in easily. The data that gets pulled out goes into those different data structures. This may seem a little inefficient but is really the optimal way to address the issue.

Linked list: the linked list is a doubly linked list that uses forwards and backwards pointers to allow me to move efficiently either way through the list.

Priority queue: this is a standard library thing that makes a que and sorts it for us probably using bubble sort. We had to overload the operator.

Heap: its basically a binary search tree with left, right, and parent nodes. This one took me several days to do.

Data:

**Results:**

The data was surprising. The heap was the fastest with the STL being the slowest. Writing stuff to files also takes a lot of time. I enjoyed this up until everything broke and I have to write this report in 5 minutes.