Yifeng Qin

CS202-1104

Christos Papachristos

11/25/17

Project9 Documentation

The purpose of this program was to understand how to use array and node queues. For the array queue, you had one big array that had a set size and you didn’t need any memory allocation for it. The node queue doesn’t have a set size so I had to give it memory. In my array queue. I had it set so that the array queue just keeps adding more to the end of the array. The mback represented the back of the queue of the certain object. The mfront had to be move so I didn’t write over the same data. So I had it set to the previous mback. The constructors were made like this so that they don’t overlap with each other. Then the front and back would return the front and back of the required queue. Push had to add one more object to the array so I had to increase the size and add it. The pop just gets rid of one element so I had to decrement it. The size was given by mback compared to front and the empty and full were just checking if size was 0 or 1000. Clear set everything back to zero and the serialize printed out the amount of data that was required for that queue. To node queue wasn’t all that different. Instead of using brackets, I had to move within the node. So I had to use the arrow operator a lot in order to get to the next node. In general they do the same things but in different implementations.

Some of the difficulties of this projects were how to use the pop and push functions. It was confusing why we had to use the module to get the back and front. But the array was pretty straight forward. The node queue was a more difficult because we had to use the arrow operators to mover around the node. But if I had to change some things to my program they would have to be making it more bullet proof. I think that the clear should be more complicated then what I am doing and that I shouldn’t set everything to zero because it may lead to a memory leak.