Kai Muramoto Lai
301434238
CMPT 225 - D100

Assignment 1 Write-Up

The functions that I tested for in this assignment are: jump(), display(), ddisplay(), eject (), reserve(), clear() in DQtest.cpp file.

The first thing I tested was my display(), and ddisplay() functions. I did this by printing out my elements into a queue. I used the given enqueue() function to print out the elements into my displays. Working on the display() and ddisplay() first really helped me in this assignment as I was able to get a visual representation of my implementation. This helped me debug my code and fix certain errors that came up while I was working on my other functions throughout the assignment. Later on into testing, I added cout << "front index=" << front << ", back index=" << back << endl; to indicate the front and back index for the circular queue which also helped me visualize my code.

The next thing I tested was my jump() function. I first made sure it was able to jump the queue and then insert a value to the front of the queue. I did this by keeping track of the front of the circular array at all times. As intended it worked skipping the queue and inserting a value to the front.

The third thing I tested was my clear() function. This function is meant to remove all the values in the queue and reset the size, capacity, front index, and back index to their initial state. When called, it cleared the circular queue and worked since I got the output < >. Additionally, I tested my clear() function alongside the reserve() function to test for edge cases. As intended it cleared the values in the circular queue and allocated space and set the queue back to its original values.

Next I tested my eject() and reserve() functions. I first made sure the function eject() was working properly by creating a small test program. Then I made sure it worked when it was called multiple times. I called it twice and it worked by removing the last element on the first call and removing the last element again on its second call.

I then tested my reserver() function. The purpose of this function is to change the capacity to a new capacity only if it is larger than its current size. I did this by calling reserve(8). By doing so it allocated space for 8 elements in the queue. Additionally as stated before I tested it alongside clear() and it worked as intended.

The last thing I tested for was other inputs to see if my program could handle other data types. I tested for a string value and it worked in my queue. Additionally, I tested my eject(), jump(), clear(), and reserve() on string values and it worked as well.