Test Results

QueueTester Output:

Queue is Empty

Queue is Empty

Sorry Queue is Full

Sorry Queue is Full

Sorry Queue is Full

Sorry Queue is Full

Queue: Walk Dog, Eat, Play Oboe, Take Out Trash, Sleep, Eat.

Priority Queue: (7, Eat), (7, Eat), (6, Sleep), (3, Play Oboe), (2, Take Out Trash),

(1, Walk Dog).

Queue: Eat.

Priority Queue: (1, Walk Dog).

Queue is Empty

Queue is Empty

The main goal of this assignment is create an object Queue that’s stores tasks in a first in first out order. I decided take on the extra challenge of using a linked list rather that an array. The Task class is simply an object that stores a task description and a priority for that task. My Queue class has an inner class Node which I use to implement my singly linked list with a front and back pointer. Queue allows the user to enqueue (add a task to the queue), dequeuer (remove the front-most task), or display the current queue. Priority Queue is a subclass of Queue which overrides the enqueuer and display classes. Its enqueuer takes the priority of each task into account and add them to the queue in order rather than at the end. Most of my issues had to do with small interactions with my Nodes and linked list. For example I originally was setting some objects equal when really I wanted to copy them, so I created a second Node constructor to handle this. One interaction I learned is java will automatically take care of objects which are no longer referenced. If I were to do this again I would like to clean up my methods and maybe create some more helper functions so that my programs would look cleaner and shorten in length.