Overview:

The application we built is a simple hot dog stand which gives the user the ability to purchase hot dogs, along with chips and drinks. The main menu of our application consists of the following options:

1. Take order
2. Display current order
3. Take payment
4. Serve order
5. Display all menu items
6. Exit

The goal is to demonstrate the implementation of queue data structure.  A queue is a linear structure that follows a particular order in which the operations are performed. The order is First In First Out (FIFO). A good example of a queue is any queue of consumers for a resource where the consumer that orders first is served first. The difference between stacks and queues is in removing items. In a stack, the item that is added last is removed first; in a queue, the item that is added first is removed first. We implemented a link-based queue. Instead of using enqueue() and dequeue() to add items to and remove items from the queue, we used add() and pop().

Implementation:

* The user initiates an order by selecting the number of hotdogs, beverages, and chips to be ordered.
* Each selection is stored in a separate list.
* A node of all 3 lists is created and pushed into a linked queue.
* When the user wants to display the information for the current order, the peek() function displays the data stored in the current node
* When an order is served, the node is removed from the queue and the next node in the queue becomes the first item in the queue.