

1. Based on what we know about linked lists, stacks, and queues, design a queue data structure:

1. What functions are we likely to need for a queue to function like the one discussed in class?

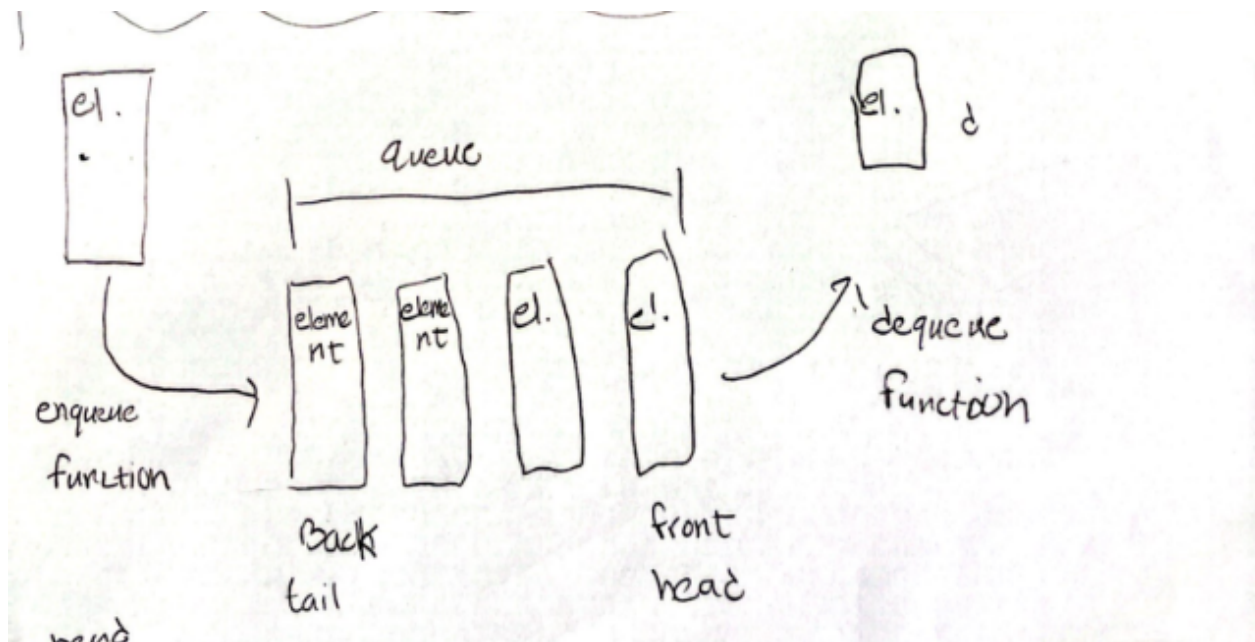
Our two main functions that are necessary for a queue data structure is the enqueue and dequeue functions. The enqueue function adds a new element at the end of a queue while the dequeue function removes an element in the front of the queue. While there are some ideal functions to create more mobility for the data structure, only the enqueue and dequeue functions are necessary to create a simple data structure.

2. What values will we need to know about the structure for our queue to function properly?

We will need to keep track of the number of element in the queue and store it into the queue. This can be done through linked lists, stacks, and arrays.

We will also need a "head" and "tail" value to keep track of the front and back of the queue. A queue data structure is like a line. The first person in the line is the "head" of the queue and the last person is the "tail" of the queue. This will help the data structure keep track of how big the queue is.

Design:



2.

Based on what we know about linked lists, design a list data structure that allows us to add (insert) or remove (delete) values at a given location in the list (instead of the top of a stack or the front or back of a queue):

1. What functions are we likely to need for a list to function like this?

Functions required for a list data structure are the insert(), delete(), and get(). We will need the insert() function to insert a value. The delete() function is required to delete the value. While these are the necessary functions to create a list data structure, to fulfill our requirements for this assignment, we will need the get() function as well. This will allow us to insert and delete a value at any location.

2. What values will we need to know about the structure for our list to function properly?

We will need the number of elements of the list. This is so we can store the element and a linked list or array can do this.

We will also need the size of the list. This is to keep track of the number of elements in a list and we will be able to insert and delete at any location since we know their location.

Lastly, we will need references to traverse the list gracefully.

Design:

