CS 220 Software Design II

Spring 2022

In-class Exercises 10

University of Wisconsin - La Crosse

Date: April 29

1. As we saw in class, a queue is represented by a List underneath the hood. Of singly linked list, doubly linked list, and array list, which list structure is best from a runtime perspective for representing a queue and why?

2. In lab, we used an array (not an array list) for implementing a stack. Using an array to represent a queue is somewhat more difficult. Why? How might we achieve this in the most efficient way possible? What attribute(s) would you need in the class? What would the methods look like for enqueue, dequeue, and front? Hint: Do not move elements already in the queue.

3. What is displayed to the console after running the code below? What does the stack looks like over time?

```
LinkedList<Integer> stack = new LinkedList<>();
   stack.push(3);
   for (int i = 1; i <= 5; ++i) {</pre>
        if (peek() % 2 == 0) {
5
            stack.push(i);
6
        } else {
7
            int r = stack.pop();
8
            stack.push(i + r);
9
        }
10
   while (!stack.isEmpty()) {
11
        System.out.print(stack.pop() + " ");
12
13
   }
```

4. What is displayed to the console after running the code below? What does the queue looks like over time?

```
LinkedList<Integer> queue = new LinkedList<>();
   queue.add(3);
   for (int i = 1; i \le 5; ++i) {
       if ((i + queue.peek()) % 2 == 0) {
5
            queue.add(i);
6
       } else {
7
            int r = queue.poll();
8
            queue.add(i + r);
9
       }
10
11
   while (!queue.isEmpty()) {
       System.out.print(queue.poll() + " ");
12
13
```

5. Explain how you could use a stack and a queue to determine if the characters in a String form a palindrome. Note: This solution is not the most efficient way to solve this problem. The problem is an exercise in understanding stacks and queues.