Juhwan Lee

CS-163

July 21st 2020

Program #2 Efficiency Write Up

The second programming assignment was about creating two different ADTs one is queue and the other is stack. The purpose of the program was to keep track of everyone in line and to hold alerts as they are sent out from the store, restaurant, or doctor's office to let people know when it is their turn. Queue was used to keep track of everyone in line because it is First In First Out structure and stack was used to hold alerts because if a person does not respond within a given amount of time, then the next person should be alerted and they will get priority (Last In First Out). The queue was implemented using a circular linked list where the rear pointer points to the last person in line, and rear->next points to the first. Circular linked list is useful for implementation of queue because we do not need to maintain two pointers for front and rear if we use circular linked list. We can maintain a pointer to the last inserted node and front can always be obtained as next of last. The stack was implemented using a linear linked list of arrays where each element in the array is a person being alerted. The array was dynamically allocated and each array is same size. These two different ADTs were implemented as a separate class and the class interfaces are in two separate .h files. In class queue, there are six functions including default constructor and destructor and those are enqueue, dequeue, peek, and display. Enqueue function adds new data to the queue, dequeue function removes the first data from the queue, peek function displays the first data of the queue, and display function displays the whole queue. In class stack, there are also six functions including default constructor and destructor and those are push, pop, peek, and display. Push function adds new data to the stack, pop function removes the last data from the stack, peek function displays the last data of the stack, and display function displays the whole stack. The information kept in the queue of people in line are person's full name and

the contact information such as cell phone number. The information kept in the stack of people alerted as to their turn are person's full name and information about when the alert took place. In main.cpp, which is the test program, I added 8 choices to test stack and queue. Here are the choices.

1. Adding new person in line
2. Alert person
3. Remove alert
4. Display next turn
5. Display alerted person
6. Display everyone in line
7. Display list of alert
8. Quit

It repeats over and over until the user choose 8 which is quit. Adding new person in line triggers enqueue function. Alert person triggers peek(queue), dequeue, and push function. The reason for peek(queue) is to get the information about the person next turn so that the user can type in the name for push function. The dequeue function triggers first and then the push function triggers. Remove alert triggers pop function. Display next turn triggers peek(queue) function. Display alerted person triggers peek(stack) function. Display everyone in line triggers display(queue) function. And lastly, display list of alert triggers display(stack) function. Using these 8 options, I tested every possible cases and all the outcomes were same as expected results.