Problem Definition

This is a program that simulates a customer service queue at an Apple store. The queue is a linked list of pointers where customers are entered into the list with their name, service needed and the time the customer arrived. Customers are then served on a first come first served basis. Customers may also be deleted from the list by name if they leave the store before being served.

Design

Main function

Display user input menu and take user input until 5 is entered

If the user inputs 1

Get the customer’s name, the service they need, the time they were added to queue, and the customer sequence number and add them to the queue

If the user inputs 2

Of the people still in queue, remove the first person entered into the queue

If the user inputs 3

Search for a user by name and remove them from the queue if found, if they are not found print a message that says not found

If the user inputs 4

Print out all of the customers currently in queue, the top of the queue being the newest customers

If the user inputs 5

Quit the program

Implementation

I wrote the program in Visual Studios Community version 12.7.6. I work on a Windows 11 computer running an Intel iCore-7 with 16 Gigabytes of RAM. The program was tested by going through the test cases listed in the program guidelines, as well as tested invalid inputs anywhere inputs could be taken.

|  |  |
| --- | --- |
| **Issue** | **Solution/Explanation** |
| After getting a user input for either the menu selection or the input name for the second the program would loop infinitely as the input buffer had not been cleared. | Every time I work with the input stream I run into an issue like this, but in essence I was getting an input from a cin statement and later using a getline() call which would eat the newline character infinitely as I did not call cin.ignore() after the cin statement to clear the input buffer. |
| Getting the current system time, but the Visual studios would flag it as unsafe. | I found quite a few different ways to get the current system time (<chrono>, <ctime>, <time.h>), but Visual studios had an issue with most of them as they all used the localtime() function which it deemed unsafe.  It did propose some alternatives including like localtime\_s() (which I ended up using). However, I had a hard time working through the documentation on how to use these safer functions. |
| Defining the static variable to use for the customer sequence | In the book reading the static variable was always defined in the header file (in examples where the class was using all inline functions) so I kept trying to define it there but to no avail. I eventually figured out it had to go in the cpp file. |
| Interacting with the static variable | Initially I thought I had to use static functions to make changes to and return static variables which was not true. |
| The user inputting a non-int into an integer variable | This can be solved by checking to see if the input given has put cin into an error state by calling cin.fail() and clearing the input stream until a valid input is given |