**Programming Project Report**

Name: Edgar Alcocer

Date: Sept - 16 - 2022

**Academic Integrity Statement:** I pledge that I have neither given nor received unauthorized help on this programming assignment.

**Problem Statement:**

The goal of this programming assignment was to demonstrate my skills using linked lists in C++. The goal of this programming assignment was to create a linked list that could be used to store information about the users in each group. The program requires several inputs that the user must write. To create the user node the user must fill out their first name, last name, username, phone number, and their message count which as default would be 0. Next, we want to add each node into a linked list, to do this the listClass is created to add new users to the list. The listClass provided the user with functions that can be called to look up information or add information to a user. incrementCount(), largestCount(), and print(). The main program begins by creating a linked list, a menu is then displayed using a do-while loop. This is allowing the user to input information and compare it with a switch statement that decides on which function to run. For error handling I checked to the menu selection for errors that are < 0 or > 6, I have also implemented a fail check point when the user does not type an int into the selection input.

**Design:**

My design for this project was to split so that main.cpp oversees creating the user list and providing the menu where the user can input data to the user node. The data structure that I have used to create userList is a linked list, linked list provide a dynamic concept of data for your users. Each userNode has all their information along with a pointer which points to the next userNode.

The algorithms I have used, userList function such as addUser() begins by going through the linked list until the tail is reached then the program inserts a new node at the end of the list. If the user inputs the same information as another user, the program will still take in the data but will only pay attention to first instance of the information. incrementCount() takes in a phonenNumber parameter that is given when the user inputs the users phone number and goes thorugh the list until the userNode’s phone number is equal to the users input, if so the message count is incremented once. After incrementCount() is called groupCount() function is called to add the message count to the total messages of the group. largestCount() allows the program to go through the list. While the userNode’s message count is greater than the previous userNode’s message count, replace the largest count with the current users count. once the program has looped the whole list the output is the largest message count along with the username to which it belongs to. If there is a tie between users who have the same amount of large message count, then the oldest user (user closest to the head) will be displayed on the screen.

**Implementation:**

For my implementation of the program is split between 5 files: main.cpp, userNode.h, userNode.cpp, userList.h, userList.cpp. I based code of the studentn2 file in csce 2014 class website. Starting with skeleton methods allowed me to easily track the pointers that need to be used in the various functions. Once the function implementation was set in the userList.h and userList.cpp file, I can begin creating a menu interface that calls methods based on user input.

**Testing:**

To test this program, I first had to figure out a well-made menu interface to even call the methods in the first place. To do this I set my menu in a do-while loop to repeat after every user input in finished. If the user inputs a number that is not on the menu, then an error will occur and prompt the user to input a correct number next time around. To test incrementCount(), phoneNumber is passed a s parameter in the function as this allows a value to be compared to it in order to decide which user should have their message count incremented. For the sample input/output it is written right under conclusions

**Conclusions:**

The overall results of this assignment were a success. Learning to use linked list has shown me the speed and optimization of searching algorithms. The project took around a week to complete, the implantation of the userList functions took the most amount of time. Something I would change differently would be to use a binary search instead of a linear search this could cut the search time in half because the program would be going through half of the list instead of all of it.

**Sample Code:**

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 1

Please add a user by First name, last name, user name, phone number, and message count

Sam Smith ssmith 4795551234 0

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 1

Please add a user by First name, last name, user name, phone number, and message count

Jim Jones jjones 4795552345 0

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 1

Please add a user by First name, last name, user name, phone number, and message count

Mary Miller mmiller 4795553456 0

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 1

Please add a user by First name, last name, user name, phone number, and message count

Tom Tompson tomtom 4795554567 0

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 1

Please add a user by First name, last name, user name, phone number, and message count

Betty Brown bbrown 4795555678 0

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 1

Please add a user by First name, last name, user name, phone number, and message count

Edgar Alcocer Castillo 3017424211 0

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 4

First Name: Sam

Last Name: Smith

User Name: ssmith

Phone Number: 4795551234

Message Count: 0

First Name: Jim

Last Name: Jones

User Name: jjones

Phone Number: 4795552345

Message Count: 0

First Name: Mary

Last Name: Miller

User Name: mmiller

Phone Number: 4795553456

Message Count: 0

First Name: Tom

Last Name: Tompson

User Name: tomtom

Phone Number: 4795554567

Message Count: 0

First Name: Betty

Last Name: Brown

User Name: bbrown

Phone Number: 4795555678

Message Count: 0

First Name: Edgar

Last Name: Alcocer

User Name: Castillo

Phone Number: 3017424211

Message Count: 0

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 2

Please input the phone number of the user you want to increase the count

4795554567

The group message count is 1

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 2

Please input the phone number of the user you want to increase the count

4795554567

The group message count is 2

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 2

Please input the phone number of the user you want to increase the count

4795554567

The group message count is 3

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 2

Please input the phone number of the user you want to increase the count

3017424211

The group message count is 4

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 2

Please input the phone number of the user you want to increase the count

4795552345

The group message count is 5

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 4

First Name: Sam

Last Name: Smith

User Name: ssmith

Phone Number: 4795551234

Message Count: 0

First Name: Jim

Last Name: Jones

User Name: jjones

Phone Number: 4795552345

Message Count: 1

First Name: Mary

Last Name: Miller

User Name: mmiller

Phone Number: 4795553456

Message Count: 0

First Name: Tom

Last Name: Tompson

User Name: tomtom

Phone Number: 4795554567

Message Count: 3

First Name: Betty

Last Name: Brown

User Name: bbrown

Phone Number: 4795555678

Message Count: 0

First Name: Edgar

Last Name: Alcocer

User Name: Castillo

Phone Number: 3017424211

Message Count: 1

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 3

The user with the highest message count is tomtom

With a total of 3 messages

Welcome to GroupMe Messaging!!!

<><><><><><><><><><><><><><><><><><><>

1. Add a user to the group

2. Increment the message count for selected user

3. Find the user with largest message count

4. Print information about all users in the group

5. To exit the program

Selection: 5

Thank you for using GroupMe messaging

Sample Code:

//----------------------------------------------

void UserList::groupCount()

{

int groupCount;

int messageCount;

string userName;

UserNode \*ptr = Head;

while (ptr != NULL)

{

messageCount = ptr->getMessageCount();

groupCount = groupCount + messageCount;

ptr = ptr->getNext();

}

cout << "The group message count is " << groupCount - 1 << endl;

cout << endl;

}