```
#ifndef MENUCLIENTE H INCLUDED
#define MENUCLIENTE_H_INCLUDED
#include <iostream>
#include <string>
#include "listacliente.h"
#include "cliente.h"
#include "nombre.h"
#include "nodocliente.h"
class MenuCliente{
private:
  ListaCliente MiLista;
public:
  MenuCliente();
  void menuPrincipal();
  void agregarCliente();
  void borrarCliente();
  void modificarCliente();
  void ordenarLista();
  void mostrarLista();
  void borrarLista();
};
#endif // MENUCLIENTE H INCLUDED
#include "menucliente.h"
using namespace std;
MenuCliente::MenuCliente() {
}
void MenuCliente::menuPrincipal() {
  char op;
  do {
     cout <<"Sistema de administracion call-center" << endl;</pre>
     cout <<"1) Agregar cliente" <<endl;</pre>
     cout <<"2) Eliminar Cliente" <<endl;</pre>
     cout <<"3)Modificar Cliente" <<endl;</pre>
     cout <<"4)Mostrar Lista"<< endl;</pre>
     cout <<"5)Ordenar la lista"<<endl;</pre>
     cout <<"6)Borrar lista"<<endl;</pre>
```

```
cout <<"Eliga una opcion: ";
     cin >> op;
     cin.ignore();
     switch(op) {
     case '1':
       agregarCliente();
       break;
     case '2':
       borrarCliente();
       break;
     case '3':
       modificarCliente();
       break;
     case '4':
       mostrarLista();
       break;
     case '5':
       ordenarLista();
       break;
     case '6':
       borrarLista();
       break;
     case '7':
       cout <<"pre>resione enter para continuar" <<endl;</pre>
       getchar();
       break;
     default:
       cout <<"Opcion invalida"<<endl;</pre>
       cout <<"pre>resione enter para continuar" <<endl;</pre>
       getchar();
  } while(op!= '7');
}
void MenuCliente::agregarCliente() {
  Nombre miNombre;
  string miCadena;
  int miEntero;
  Horario miHorario;
  Horario miDuracion;
```

cout <<"7)Salir" <<endl<<endl;</pre>

```
Cliente miCliente;
  cout<<"Agregar Cliente"<<endl<<endl;</pre>
  cout < < "Nombre de la persona" < < endl;</pre>
  getline(cin, miCadena, ' ');
  miNombre.setApellido(miCadena);
  getline(cin, miCadena, ' ');
  miNombre.setApellido(miCadena);
  getline(cin, miCadena, '\n');
  miNombre.setNombre(miCadena);
  cout < < endl < < endl;
  cout<<"Hora de atencion\nhora: ";</pre>
  cin>>miEntero;
  miHorario.setHora(miEntero);
  cout < < "minuto: ";</pre>
  cin>>miEntero;
  miHorario.setMinuto(miEntero);
  cout < < endl < < endl;
  cout < < "Duracion de la llamada\nminuto: ";</pre>
  cin>>miEntero;
  miDuracion.setHora(miEntero);
  cout < < "segundo: ";
  cin>>miEntero;
  miDuracion.setMinuto(miEntero);
  cout < < endl < < endl;
  miCliente.setNombre(miNombre);
  miCliente.setHoraDeLlamada(miHorario);
  miCliente.setDuracion(miDuracion);
  MiLista.insertar(miCliente);
void MenuCliente::borrarCliente() {
  Nombre miNombre;
  string miCadena;
  Cliente miCliente;
  cout<<"Borrar Cliente"<<endl<<endl;</pre>
  cout < < "Ingresa el nombre del cliente" < < endl;</pre>
  getline(cin, miCadena, ' ');
```

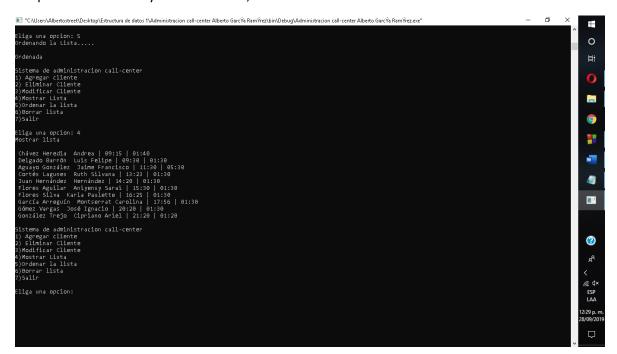
}

```
miNombre.setApellido(miCadena);
  getline(cin, miCadena, ' ');
  miNombre.setApellido(miCadena);
  getline(cin, miCadena, '\n');
  miNombre.setNombre(miCadena);
  miCliente.setNombre(miNombre);
  try{
     MiLista.eleminar(miCliente);
  }catch(ListException){
  cout<<"No encontrado, Eliminar->eliminar"<<endl<<endl;</pre>
void MenuCliente::modificarCliente() {
  Nombre miNombre;
  string miCadena;
  int miEntero;
  Horario miDuracion;
  Cliente miCliente;
  NodoCliente *aux;
  cout<<"Modificar Cliente"<<endl<<endl;</pre>
  cout < < "Ingresa el nombre del cliente" < < endl;</pre>
  getline(cin, miCadena, ' ');
  miNombre.setApellido(miCadena);
  getline(cin, miCadena, ' ');
  miNombre.setApellido(miCadena);
  getline(cin,miCadena,'\n');
  miNombre.setNombre(miCadena);
  miCliente.setNombre(miNombre);
  try{
     aux = MiLista.localiza(miCliente);
     miCliente = aux->getCliente();
     cout << "Modificacion de la duracion de la llamada del cliente: " <<
miCliente.getNombre() < < endl;
     cout<<"Ingresa la nueva duracion\nminuto: ";</pre>
     cin>>miEntero;
     miDuracion.setHora(miEntero);
```

```
cout < < "segundo";</pre>
     cin>>miEntero;
     miDuracion.setMinuto(miEntero);
     cout<<endl;
     miCliente.setDuracion(miDuracion);
     aux->setCliente(miCliente);
  }catch(ListException){
   cout<<"No hay Datos, ModificarCliente->localiza"<<endl<<endl;</pre>
}
void MenuCliente::ordenarLista() {
  try{
   cout < < "Ordenando la Lista....." < < endl < < endl;
     MiLista.ordena();
     cout<<"Ordenada"<<endl<<endl;
  }catch(ListException){
  cout < < "No hay Datos, OrdenarLista->ordena" < < endl < < endl;
  }
}
void MenuCliente::mostrarLista() {
  try{
     cout < "Mostrar lista" < < endl < < endl;
     cout < < MiLista.to String() < < endl;
  }catch(ListException) {
  cout<<"No hay Datos , MostrarLista ->toString"<<endl<<endl;</pre>
  };
}
void MenuCliente::borrarLista() {
  cout < < "Borrando la lista" < < endl;
  try{
     MiLista.eliminarTodo();
      cout < < "Borrada" < < endl < < endl;
  }catch( ListException ) {
     cout < "No hay Datos, BorraLista -> eliminarTodo" < < endl < < endl;
  };
}
```

```
Stitems de administracion call-center
1) Agragar Cliente
2) Elishinar Cliente
3) Agragar Cliente
3) Hodrigar Cliente
3) Modificar Cliente
3) Octobra I la lista
5) Octobra I la lista
6) Octobra I la lista
7) Salla
7) Octobra I la lista
7) Octobra I lista
7) Octobra I
```

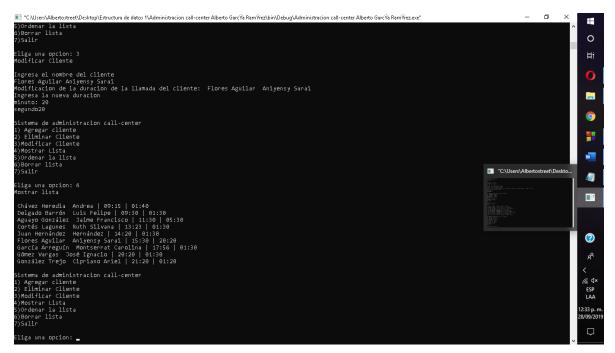
Despues de hacer la inyección de texto, mostre la lista



Ordene la lista y la volvi a imprimir, ahora ya están ordenados por la hora de llamada

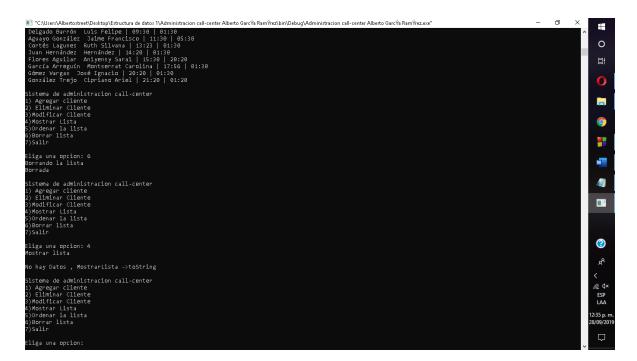
```
Chiver Alberto Desitory Estuation de data 11 Administracion cult-center Alberto Garc's Ram/wezbin/Debug/Administracion Cultor Alberto Garc's Ram/wezbin/Debug/Administracion Cultor Alberto Garc's Ram/wezbin/Debug/Administracion Cultor Cultor Alberto Garc's Ram/wezbin/Debug/Administracion Cultor Cultor
```

Para comprobar que funciona la eliminación y el cambio en la lista tome el nombre de : Flores Silva Karla Paulette que estaba mas o menos en medio de la lista y la eliminacion fue correcta.



Ahora modifique la duración de la llamada porque en el proyecto pedia la opción de editar la duración de la llamada y al volver a imprimir después de cambiar a : Flores Aguilar Aniyensy Sarai

A 20 minutos con 30 segundos de duración de la llamada.



Ahora elimine la lista y la quise volver a imprimir pero esta vez ya no me mostro la lista en cambio me mostro el mensaje del try-catch que no hay datos porque la lista se elimino satisfactoriamente.