Universidad de Guadalajara

Reporte Sistema en funcionamiento

Garcia Ramírez Alberto

Problema Planteado:

Descripción: Una empresa de fabricación de equipos de cómputo necesita un programa que le ayude a llevar un control las llamadas que sus agentes han atendido durante el día.  El programa deberá almacenar los nombres de los agentes con su respectivo horario, número de empleado, número de extensión, horas extras trabajadas, especialidad y la lista de usuarios atendidos durante su turno.

La secretaria encargada podrá agregar o eliminar agentes como sea necesario, así como usuarios atendidos por cada agente.

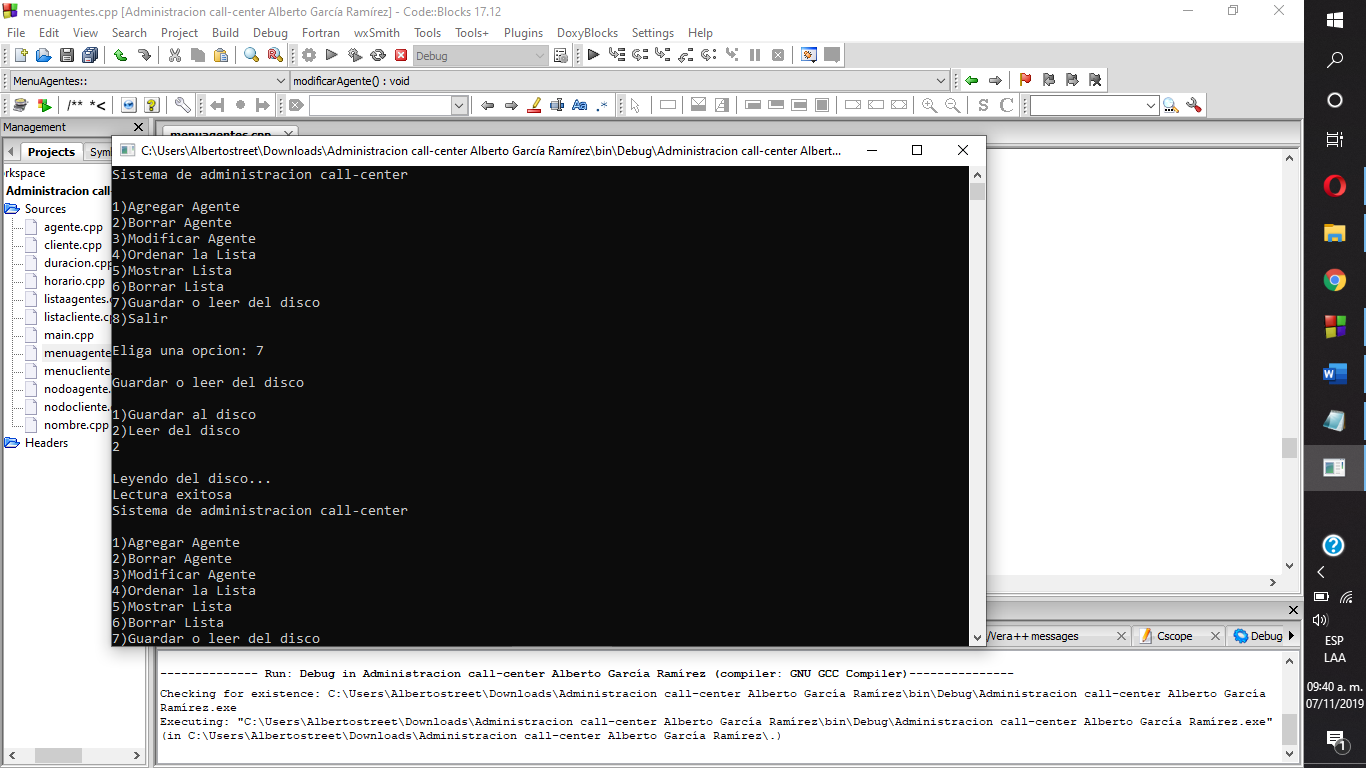
El programa contará con las funciones:

* Mostrar toda la lista de agentes almacenados, o bien filtrarlos por especialidad.
* Las especialidades para los agentes pueden ser: **Servidores, De escritorio, Portátiles, Linux, Impresoras, Redes**
* Agregar un agente
* Encontrar y mostrar los datos de un agente (número de empleado, nombre, especialidad, número de extensión, horario, horas extras, y usuarios atendidos) a través del nombre del agente y su horario
* Eliminar un agente a través de su nombre y horario
* Eliminar todos los agentes
* Ordenar agentes por nombre o por especialidad
* Agregar un cliente a la lista de un agente
* Eliminar un cliente de la lista de clientes atendidos por un agente
* Eliminar todos los clientes de la lista de un agente
* Modificar la duración la llamada de un cliente
* Guardar toda la lista de agentes y clientes atendidos en almacenamiento secundario
* Leer toda la lista agentes y clientes desde almacenamiento secundario
* Mostrar la lista de agentes con o sin lista de clientes

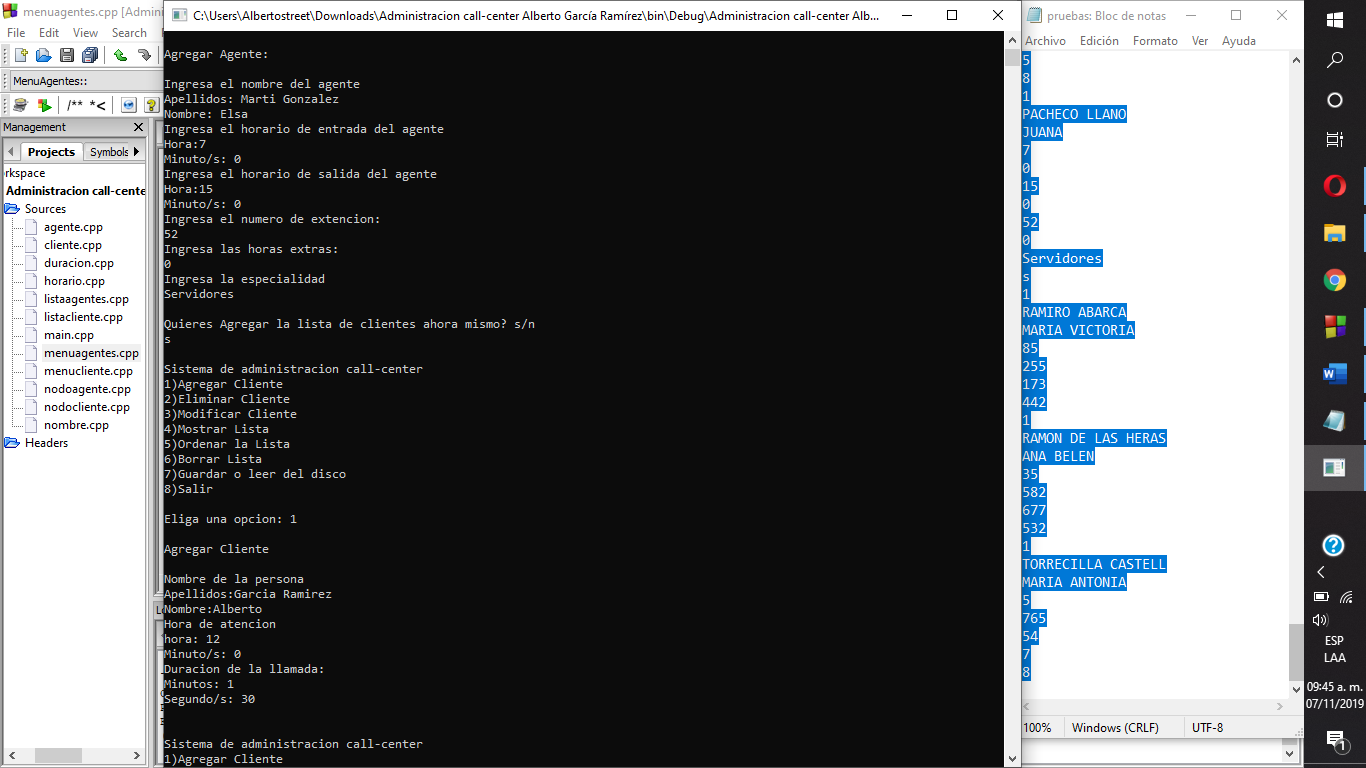
Características del programa:

* El paradigma de programación debe ser Orientado a Objetos
* El programa debe basarse en el uso de listas doblemente ligadas para los agentes, y listas simplemente ligadas para los usuarios atendidos por cada agente.
* El método de ordenamiento debe ser recursivo.
* Todos los TDA y sus funciones deben ser programadas por el alumno y contenerse en bibliotecas o colecciones bien definidas. Es decir que no habrán de utilizarse los TDA, o funciones para TDA incluidas en el lenguaje de programación elegido.

El programa debe hacerse en un lenguaje de programación portable

Menu Principal

1)Agregar Agente, debería desplegar diferentes estradas para cada dato del agente.

Con una inyeccion de texto ingresa cada uno de los datos y despliega el menu para los clientes porque se eligio la opcion en el agente de agregar los clientes en ese momento.

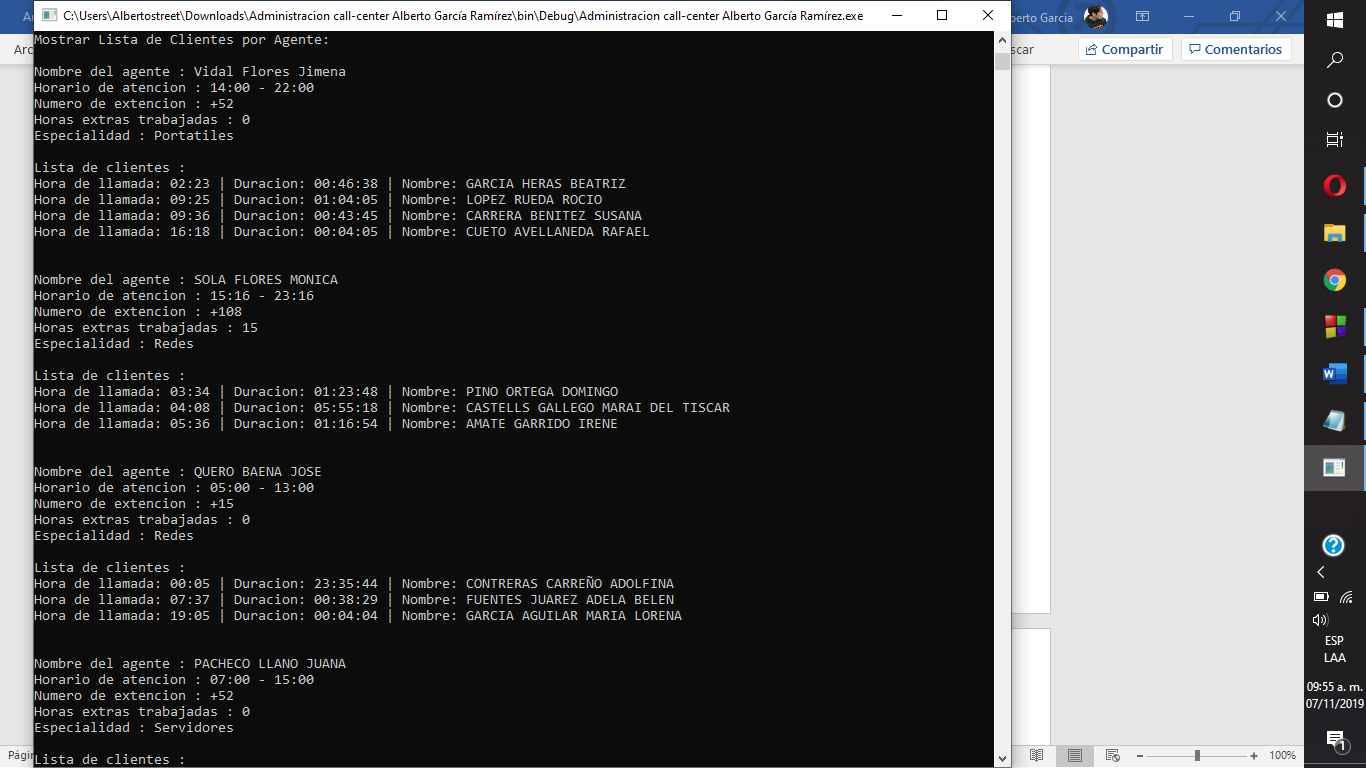
2)Borrar Agente, debería borrar el cliente de la lista junto con todos sus clientes.

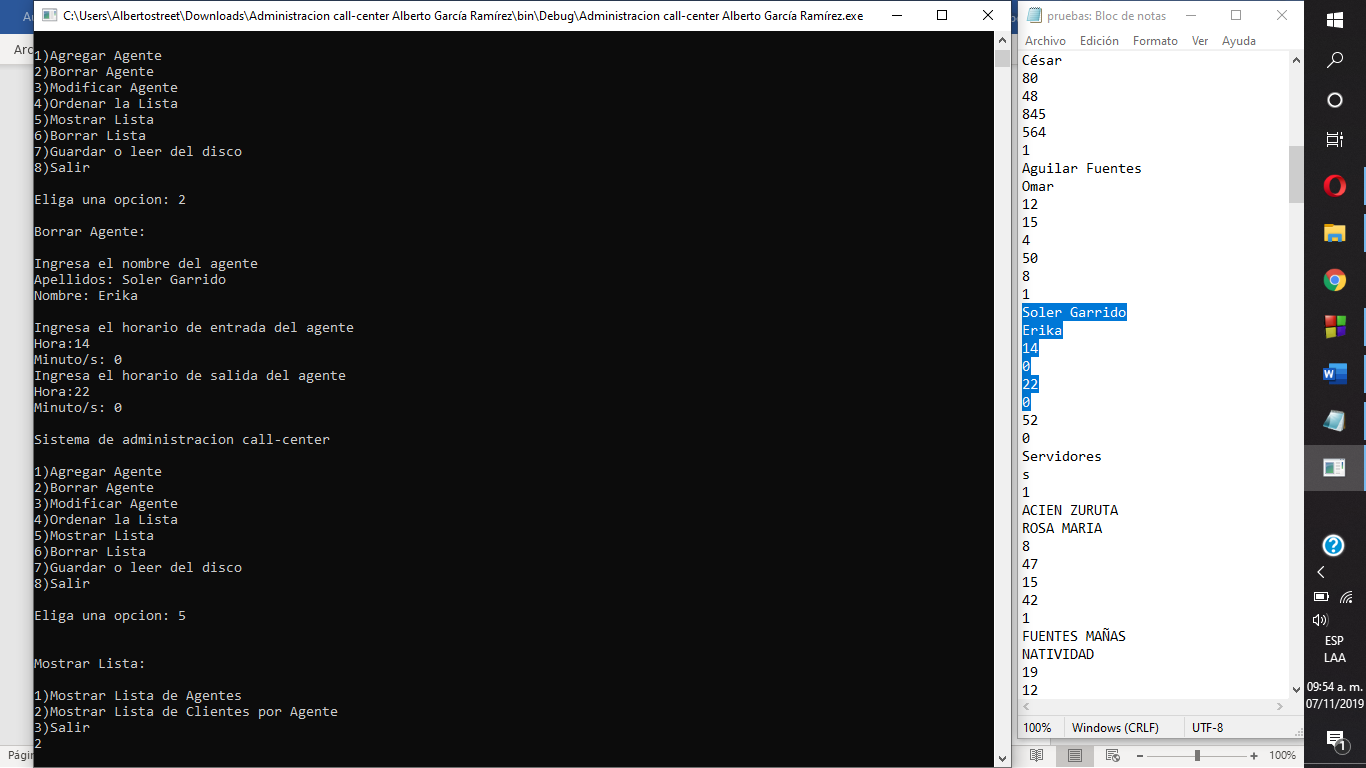
Muestro la lista para corroborar que si existe el agente con su lista.

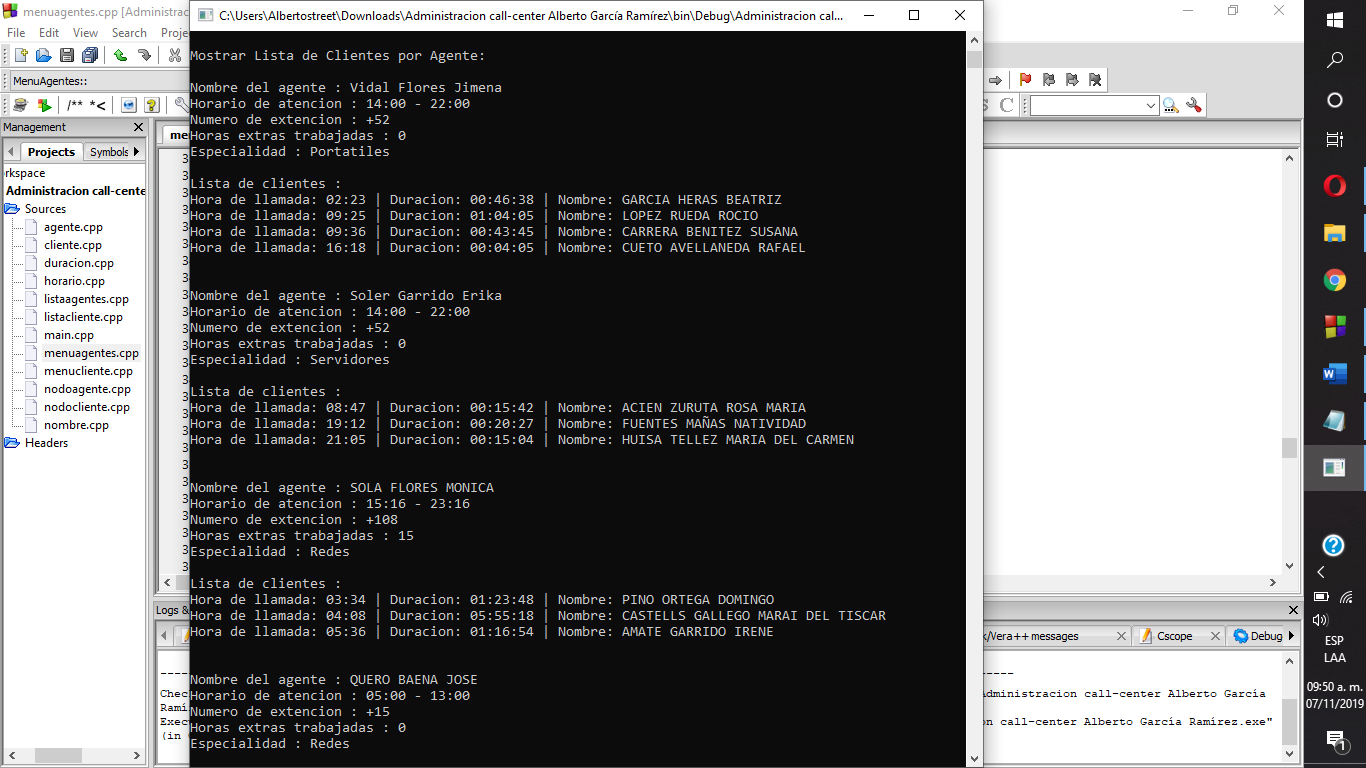
Utilizo la opción de eliminar e ingreso los datos del agente.

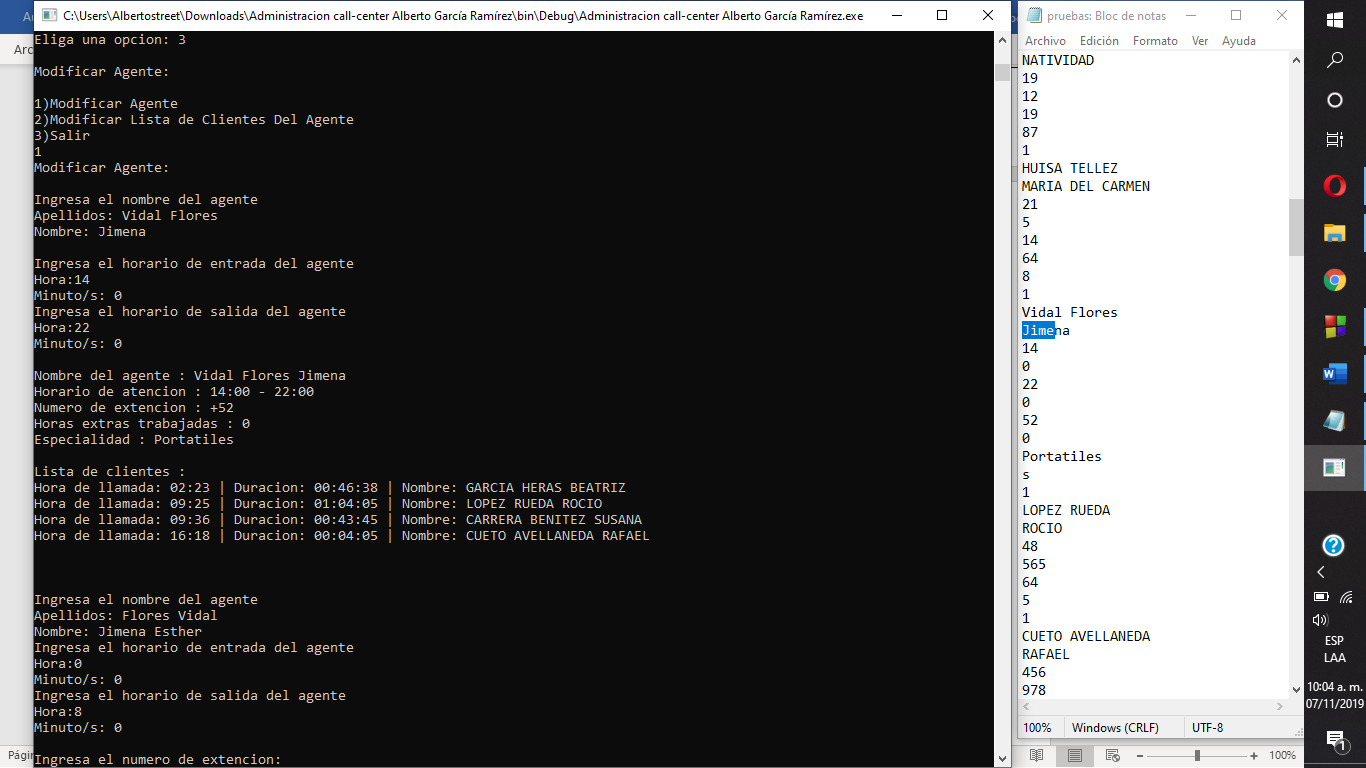
(Soles Garrido Erika)

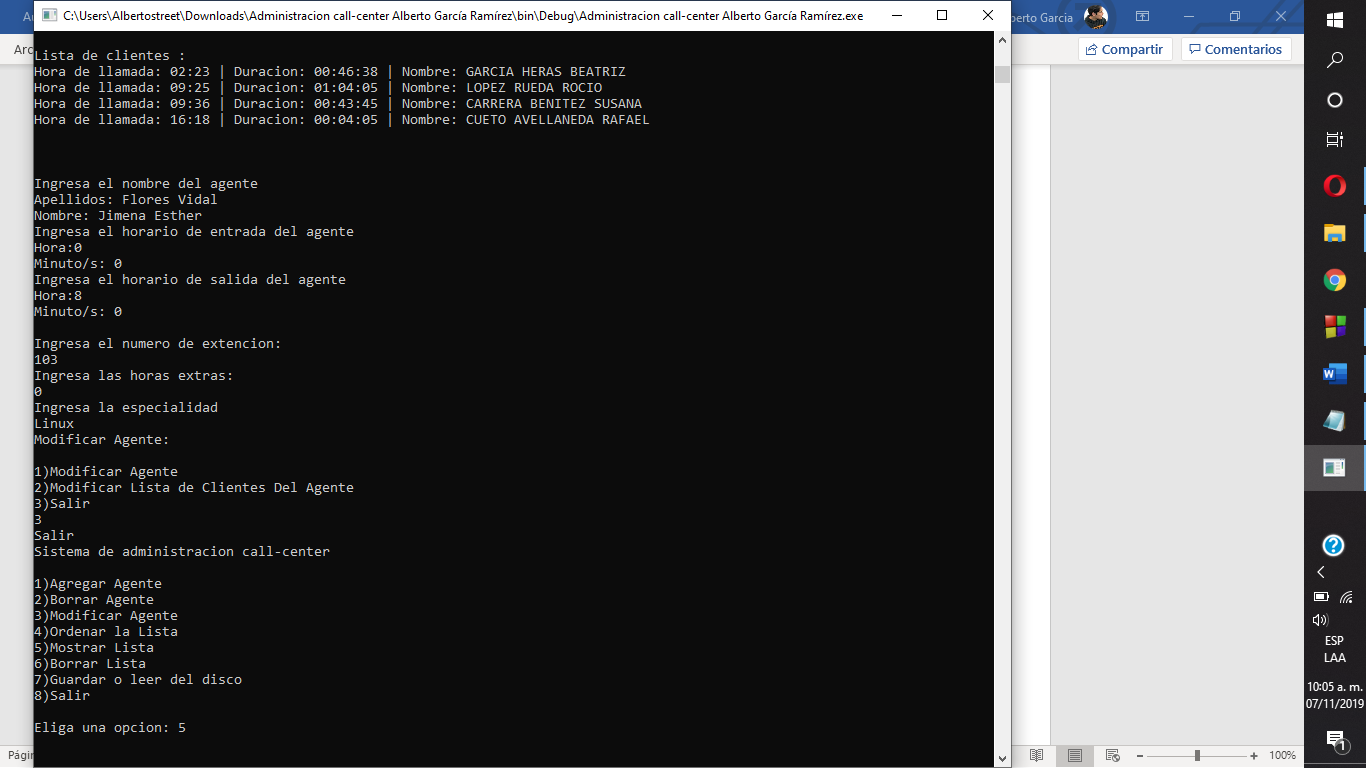
Que esta entre (Entre Vidal Flores Jimena y Sola Flores Monica)

Como se puede apreciar, ya no aparece el agente, ni su lista.

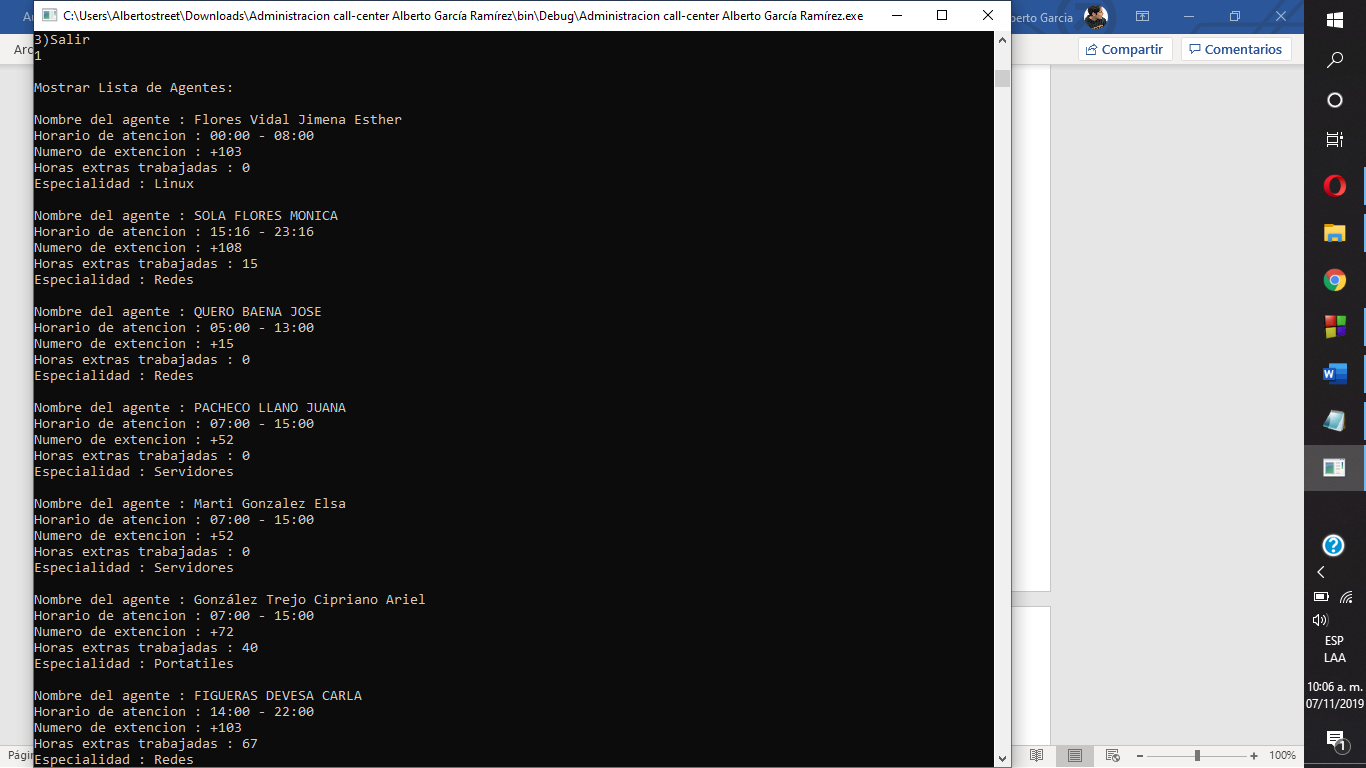
(Entre Vidal Flores Jimena y Sola Flores Monica)

3)Modificar Agente, 3.1)esta opcion despliega otro menu en el que puedes modificar los datos del agente o modificar la lista de los clientes de un agente ingresado.

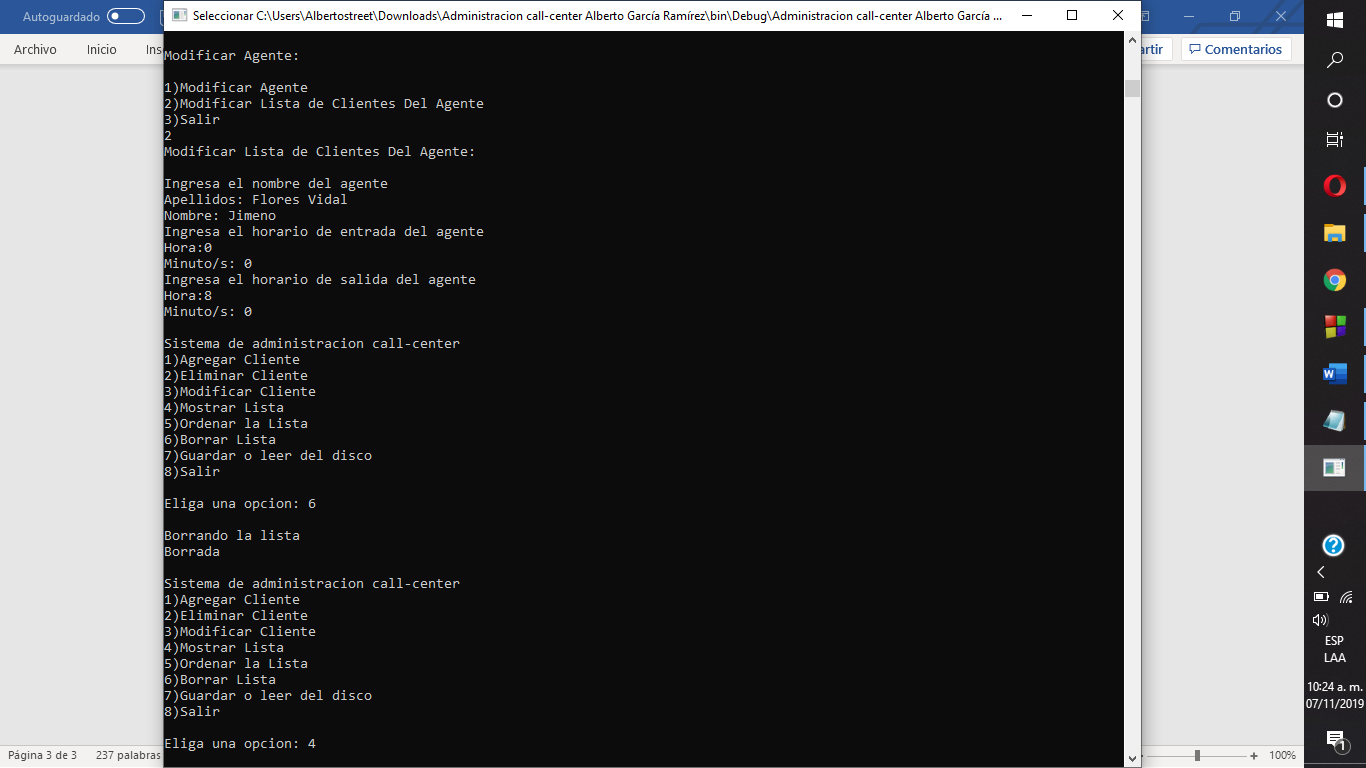
3.1) Ingreso los datos del agente y me mustra sus datos y la lista de clientes del agente, en caso de no encontrarlo, mandaria un mensaje que no lo encontro.

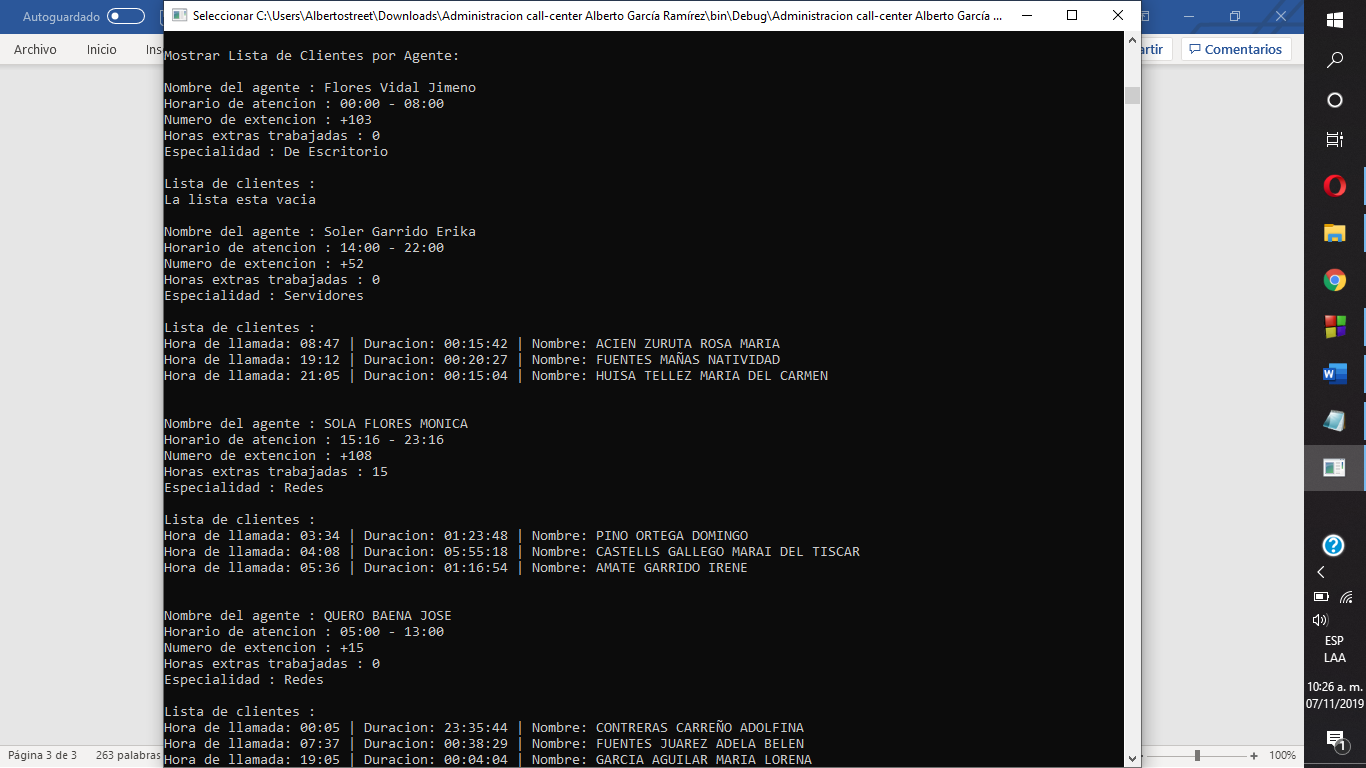


Cambio los datos del agente, sin modificar la lista por ende debería mostrar los datos nuevos y la misma lista.

El primer agente mostrado es el agente modificado con sus nuevos datos, en la misma posición en la que se ingreso porque no se ha ordenado.

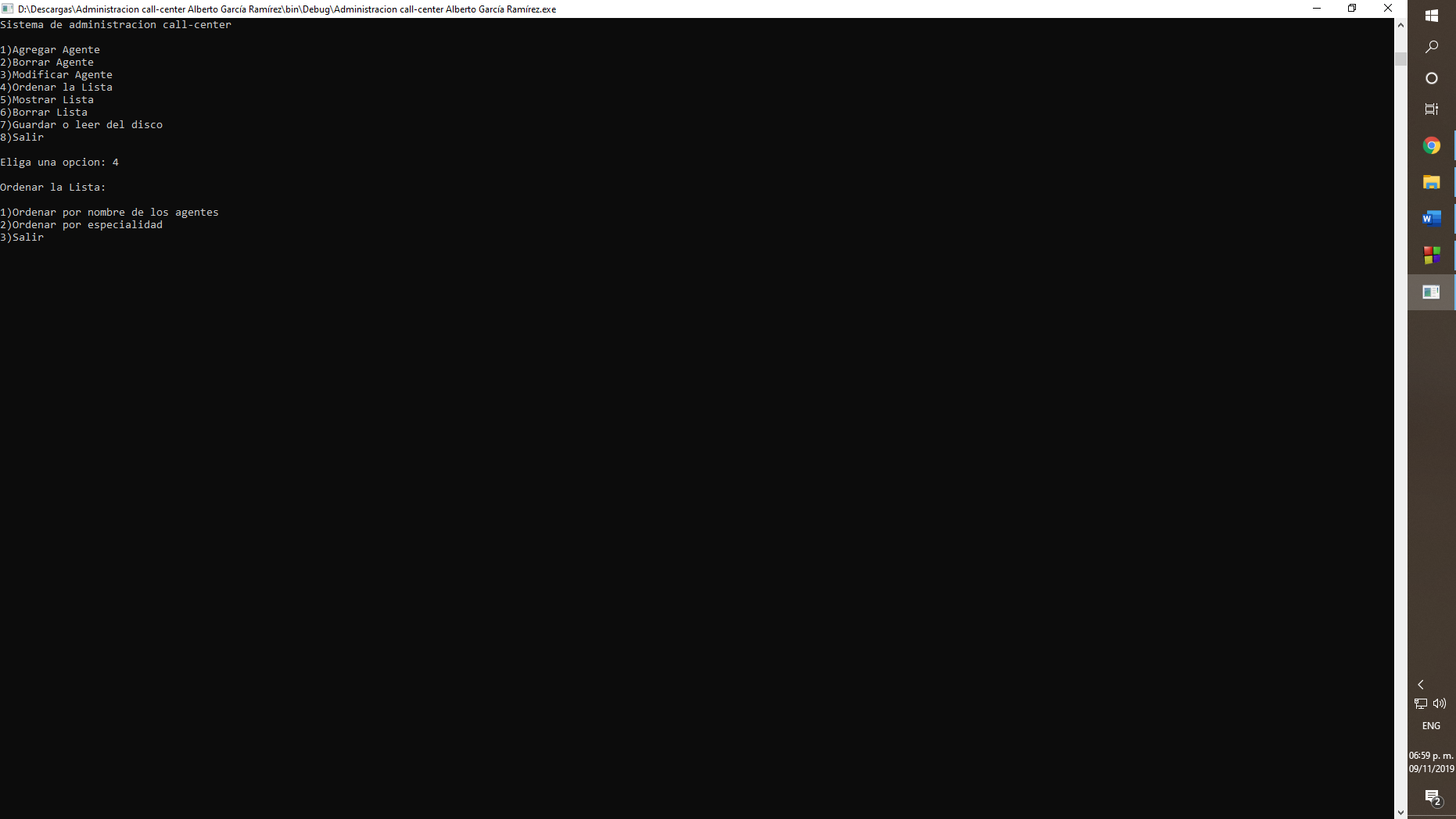
3.2) Modificar la lista de clientes del agente que se ingresa.

Ingrese un agente previamente modificado para borrar su lista, como es el mismo agente pero modificado sus datos deberia encontrar la lista que tenia y borrarla.

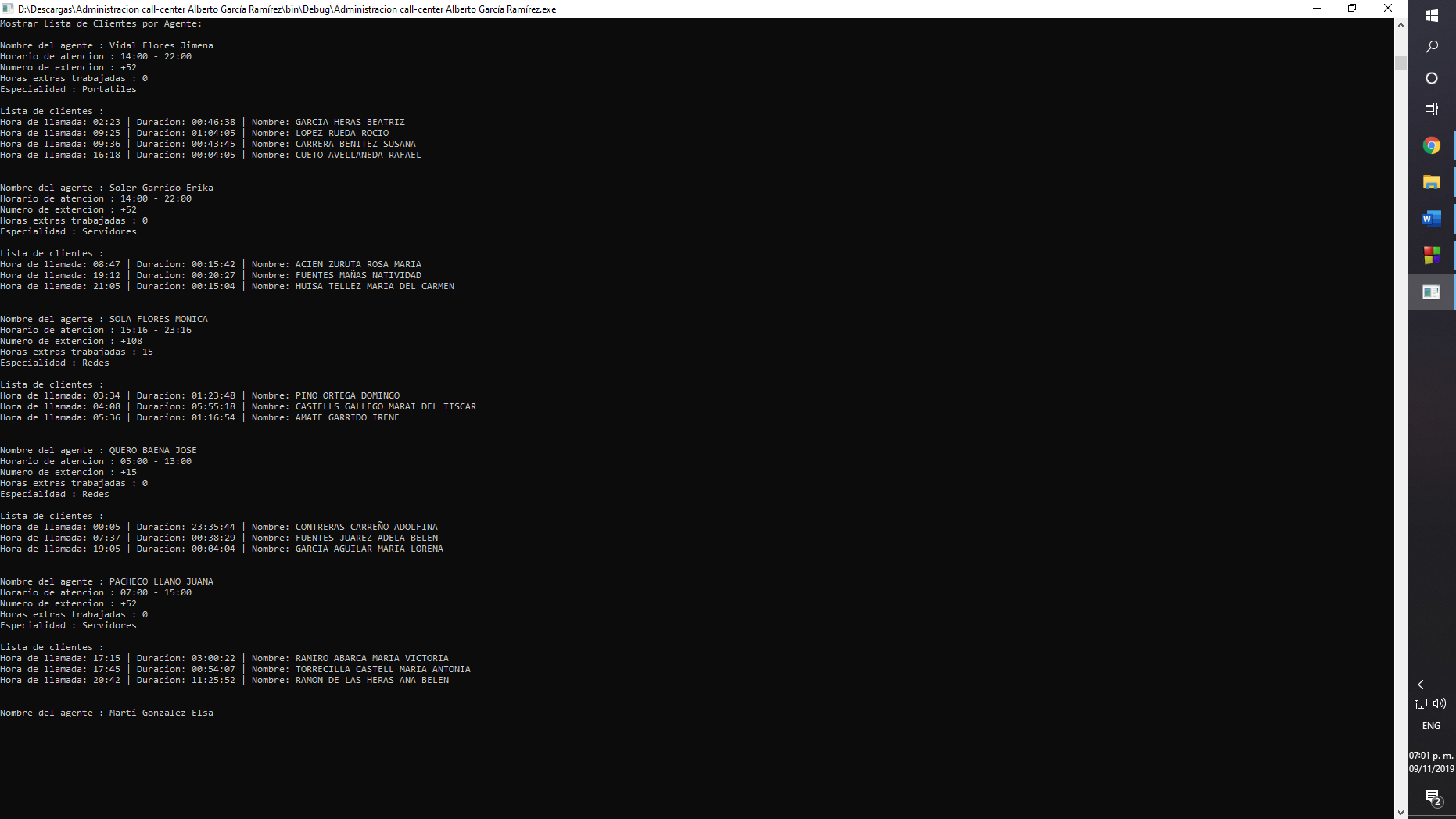


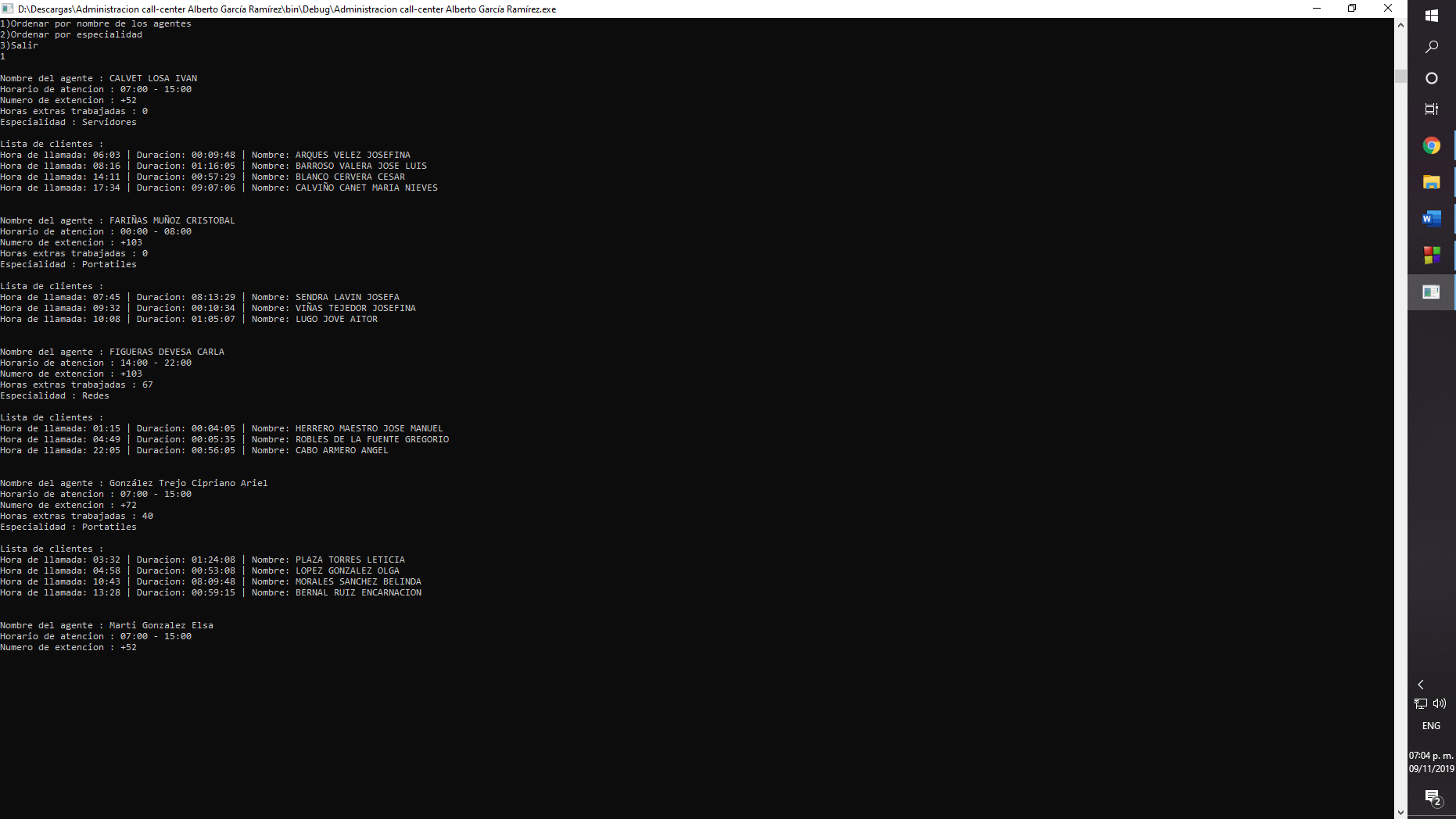
Encontró la lista, y la borro, al momento de mostrar la lista, aparece el agente modificado con la lista vacía.

4)Ordenar la lista, esta opción tiene un submenú para escoger si se ordena por nombre del agente o por especialidad, en mi proyecto al momento de ordenar los agentes también ordena la lista de clientes de cada agente por hora de llamada.

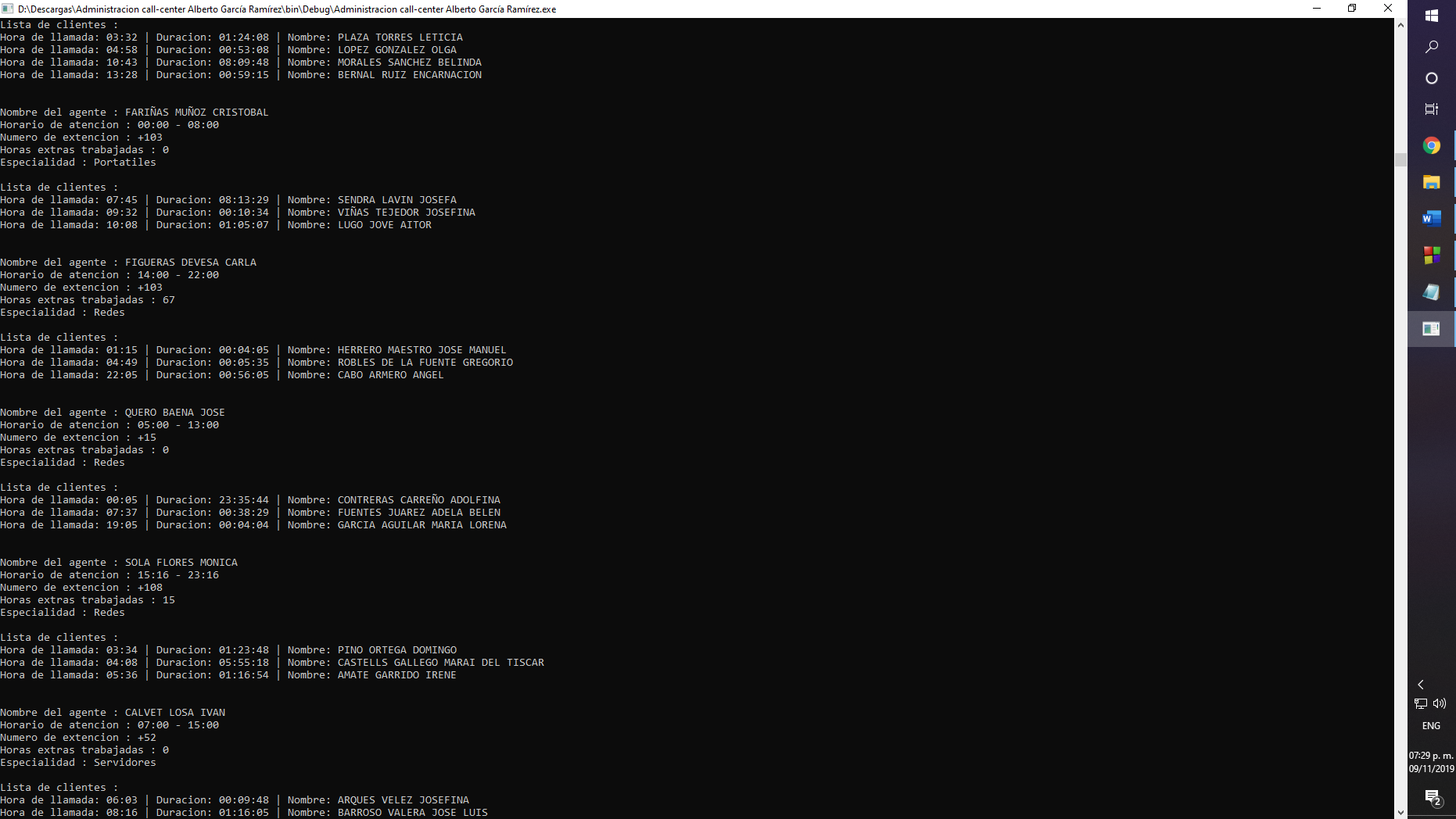
4.1) Como dice, ordena la lista de agentes por nombre en orden alfabético.

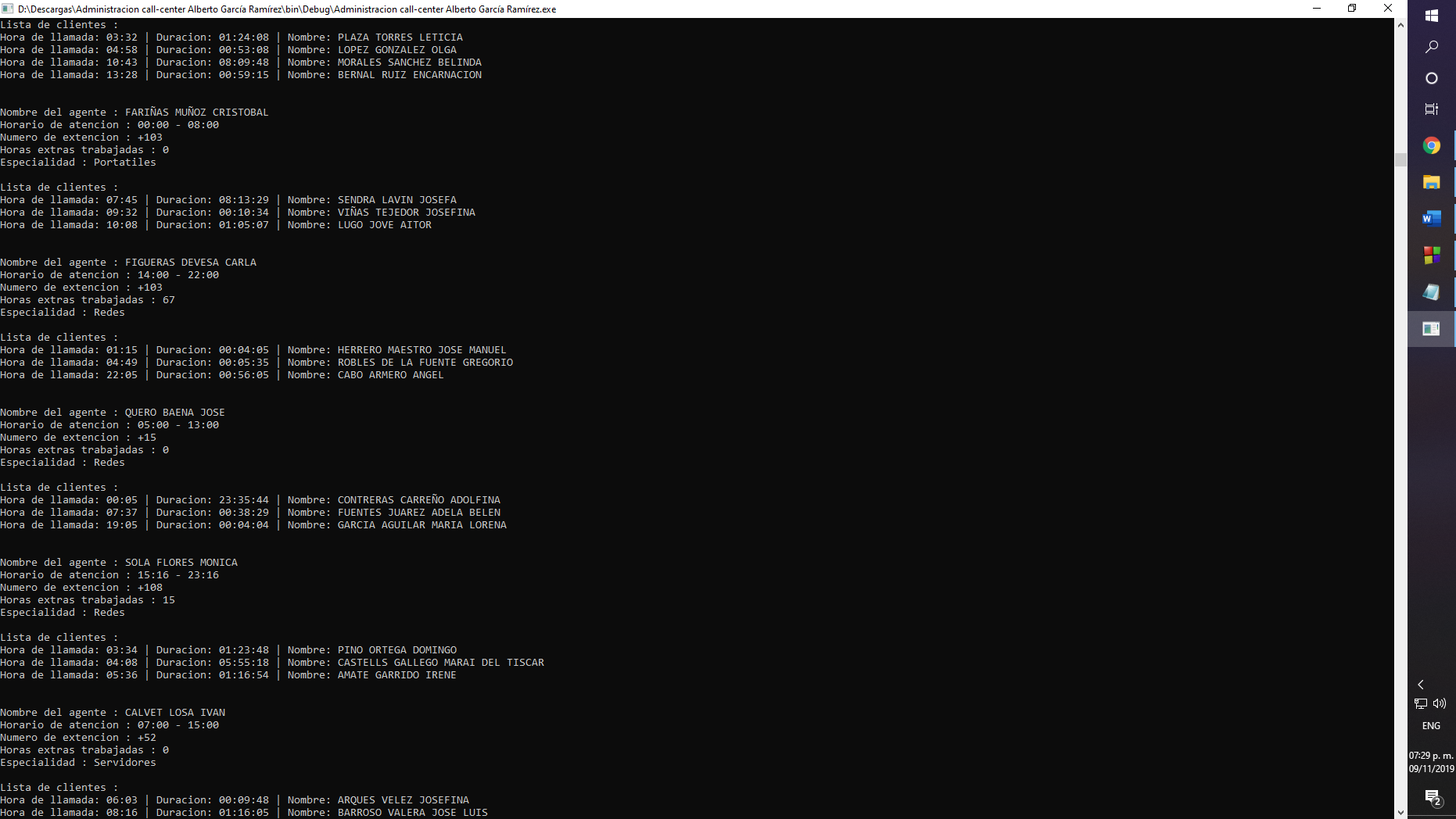
4.2) Ordena la lista por especialidad en orden alfabético.

Muestro la lista sin ordenar, agente y lista de clientes.

4.1)

4.2)



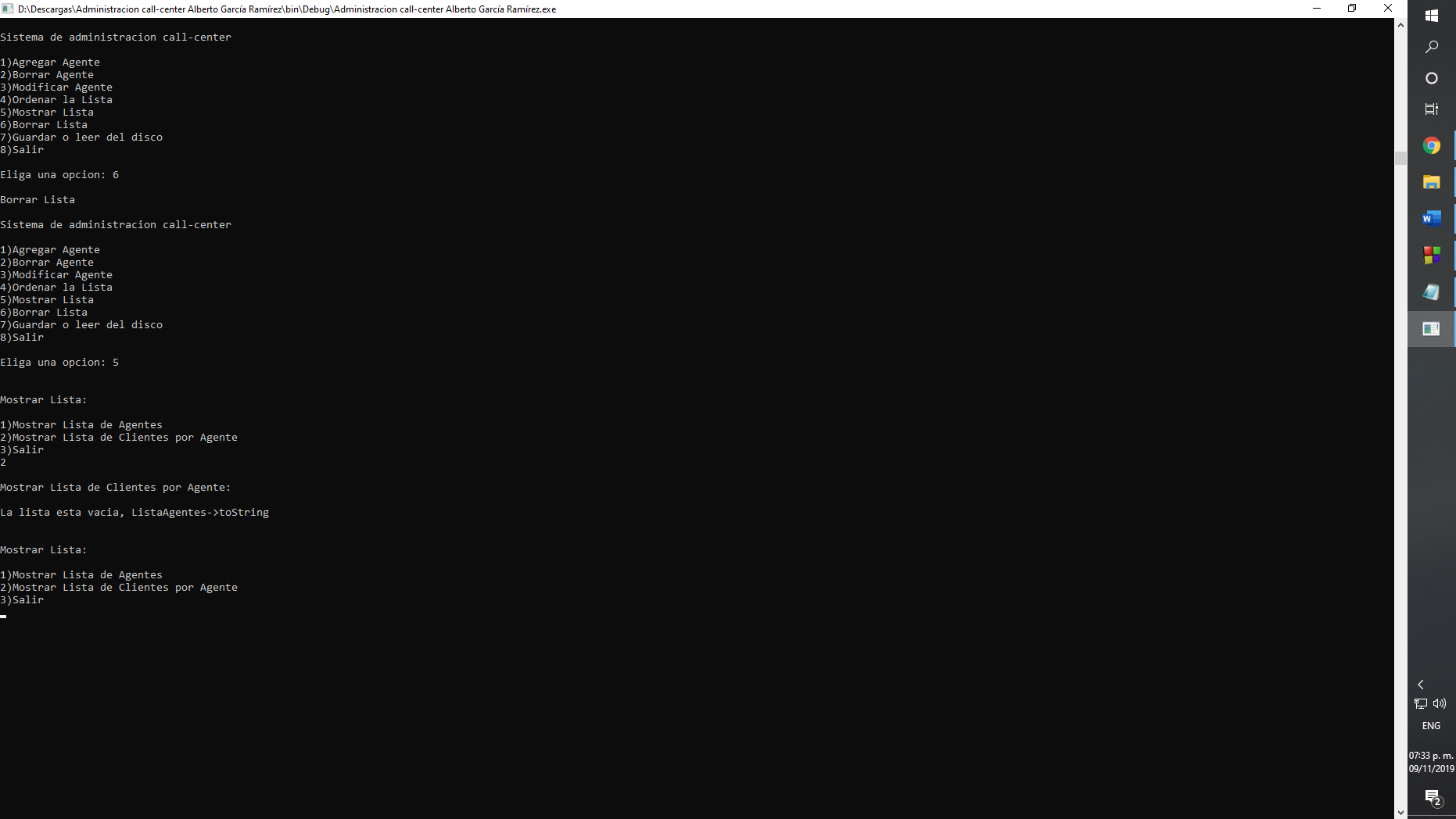


5)Mostrar lista, muestra la lista, tiene un submenu 5.1) mostrar lista de agentes, solo muestra los agentes.

5.2)mostrar lista de agentes con su lista de clientes, muestra los datos del agente y su lista de clientes.

Recorre la lista, imprime los datos del agente y su lista o no dependiendo de la opcion.

6)Borrar la lista, borra la lista de agentes y la lista de clientes de cada agente

en caso de estar vacía la lista manda una excepción.

Y en caso de querer imprimirla manda una excepción porque la lista esta vacía y no hay nada que imprimir.

7)Guardar o leer del disco, tiene un submenú

7.1) Guardar al disco, crea un archivo “Lista de Agentes.txt” que almacena la lista de agentes, pero al momento de guardar un agente, se manda guardar su lista de clientes con su “nombre(del agente).clientes.txt”, y asi sucesivamente hasta que termine con todos los agentes.

Al guardar, se guardan en una carpeta aparte del programa llamada “Archivos”.

7.2) Leer del disco, busca en la carpeta “Archivos” la lista de agentes y también busca la lista de clientes de cada agente, para al final quedar la lista de agentes y sus lista de clientes ya cargadas.

