**Universidad Tecnológica Nacional**

**Facultad regional de Tucumán**

**Ingeniería en Sistemas de Información**

**Catedra**: Algoritmos y Estructura de Datos

**Alumnos:**

* Cedrón Sabrina Gabriela
* Chacón Luciano

**Tema:** Trabajo Grupal Integrador 2do Cuatrimestre.

**Comisión:** 1K2

**Ciclo lectivo 2020**

INDICE

[**Resumen:** 1](#_Toc58861098)

[**1.** **Consiga** 1](#_Toc58861099)

[**1.1.** **MÓDULO CONSULTORIO** 1](#_Toc58861100)

[**1.1.1.** **Casos de prueba** 2](#_Toc58861101)

[**1.2.** **MODULO RECEPCION** 5](#_Toc58861102)

[**1.2.1.** **Casos de prueba** 6](#_Toc58861103)

[**1.3.** **MÓDULO ADMINISTRACIÓN** 8](#_Toc58861104)

[**1.3.1.** **Casos de prueba** 8](#_Toc58861105)

**Resumen:**

El siguiente trabajo consta del análisis y seguimiento del desarrollo de las consignas propuestas por la cátedra de Algoritmos y Estructuras de Datos, en un intento de integrar los diversos conocimientos adquiridos durante su cursado, por medio de la implementación de un conjunto de programas interconectados con menús y archivos de datos de acceso compartido. El código aquí expuesto fue escrito tratando de cumplir con las buenas prácticas de la programación que se vieron durante el año. Además, su armado fue gestionado, controlado y compartido, por medio de la herramienta de control de versiones Git y la plataforma GitHub.

1. **Consiga**

En este segundo trabajo integrador se desarrollará una aplicación que permita registrar las atenciones de las mascotas que asisten de un centro veterinario con múltiples consultorios de distintos profesionales. Deberá constar también de un módulo para la gestión de turnos y otro para la administración gerencial.

* 1. **MÓDULO CONSULTORIO**

Al inicio de este programa se declararon las librerías pertinentes para el desarrollo del mismo, además se declararon constantes las cuales nos facilitaran el nombramiento de cada uno de los archivos utilizados.

Seguidamente se declararon las diferentes estructuras, una estructura es un tipo de dato compuesto que permite almacenar un conjunto de datos de diferente tipo. Los datos que contiene una estructura pueden ser de tipo simple (caracteres, números enteros o de coma flotante etc.) o a su vez de tipo compuesto (vectores, estructuras, listas, etc.).

En un tercer punto se encuentra la función principal del programa, dentro de la cual se incluye la parte primordial del código. En **main(),** incluimos en un primer lugar la función **setlocale(LC\_ALL,”spanish”** la cual nos permite mostrar por pantalla aquellas palabras que contengan acentos y palabras con “ñ”. En primer lugar, se declaran variables numéricas, las cuales contendrán la cantidad de turnos, veterinarios, mascotas y usuarios.

A continuación, se procede a desplegar el menú de usuario, mostrando las respectivas opciones que el usuario posee.

La función **cargarRegistros** permite obtener todos los datos guardados dentro de los distintos archivos binarios a utilizar, estos datos serán guardados en las distintas estructuras antes mencionadas, se procede a extraer estos datos ya que facilitará el trabajo a la hora de manipular estos elementos.

La función **IniciarSesion** es necesaria para validad los nombres de usuarios y contraseñas ingresadas con los datos que poseea el archivo binario Usuarios.dat, mediante estructuras repetitivas se recorrerá cada registro buscando coincidencias, cuando encuentre el nombre de usuario y contraseña buscado, finalizará la búsqueda y le permitirá al usuario entrar al sistema.

La función **ListarTurnos** será necesaria para listar aquellos turnos pertenecientes a un veterinario determinado, esta función recibe por teclado la fecha correspondiente a algún turno y busca en el listado de turnos provisto por Turnos.dat algún turno que coincida, de ser correcto el sistema listará todos los turnos correspondientes para la fecha ingresada.

La función **registrarEvolucion** recibe la fecha del turno a tratar y el DNI del dueño de la mascota, una estructura repetitiva FOR buscará en cada registro coincidencias tanto como en fecha como DNI del dueño, una vez encontrada listará la historia clínica correspondiente a la mascota, la cual contiene datos de interés para el veterinario. Una vez realizado el chequeo médico a la mascota, se procederá a registrar su evolución, la cual quedará guardada en uno de los campos del registro Turnos para su posterior lectura.

Por último, la función **guardarArchivo** escribirá nuevamente los datos modificados en el archivo binario Turnos.dat (el cual fue abierto y leído al comienzo del programa).

* + 1. **Casos de prueba**

Al inicio del programa se despliega el siguiente MENU:

==============================================================

M O D U L O C O N S U L T O R I O

==============================================================

1-INICIAR SESION

--------------------------------------------------------------

2-VISUALIZAR LISTA DE ESPERA DE TURNOS (INFORME)

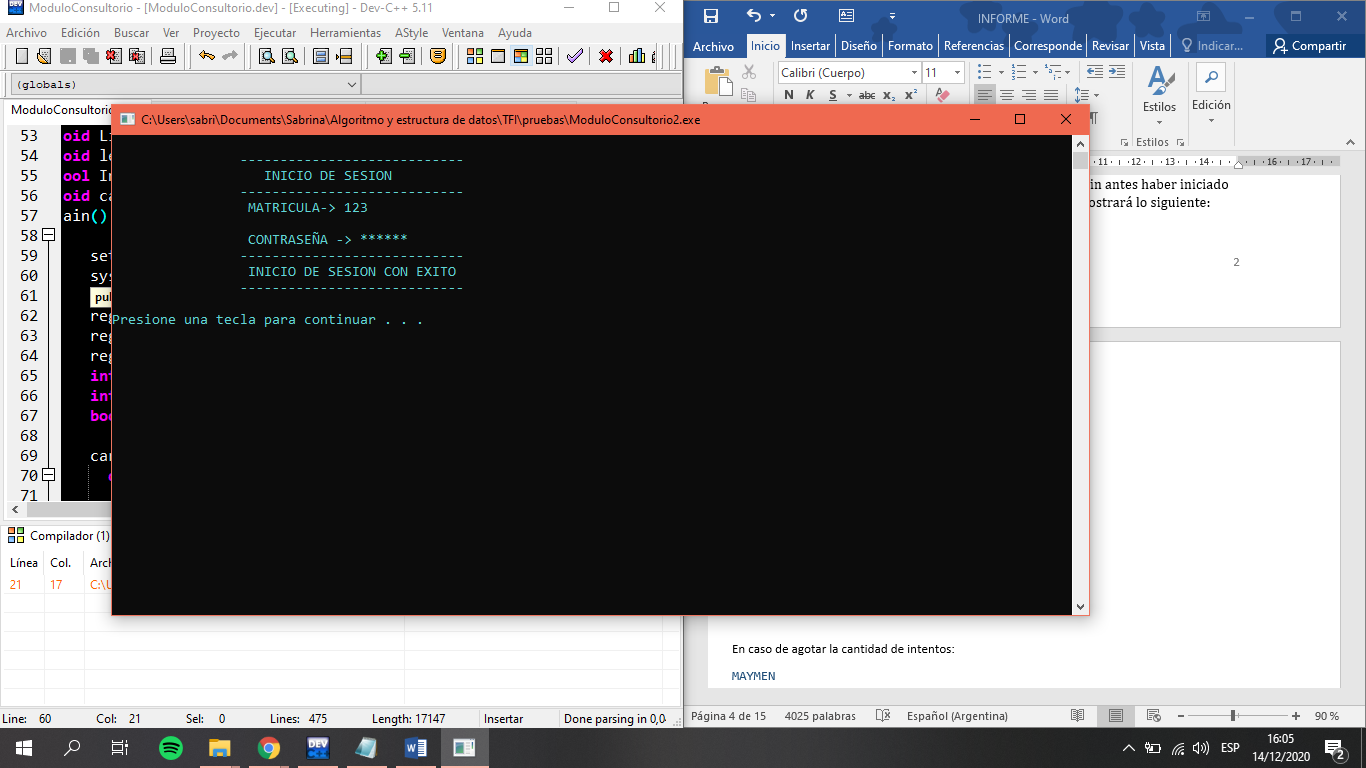
--------------------------------------------------------------

3-REGISTRAR LA EVOLUCION DE LA MASCOTA

--------------------------------------------------------------

4-SALIR DE LA APLICACIÓN --------------------------------------------------------------

INGRESE SU OPCION -->



Este programa no permitirá el acceso a las opciones 2 y 3 sin antes haber iniciado sesión. Una vez ingresada, si presionamos la opción 2 se mostrará lo siguiente:

**INGRESE LA FECHA PARA BUSCAR SUS RESPECTIVOS TURNOS...**

**DIA: 13**

**MES: 12**

**AÑO: 2020**

**=============================================================**

**LISTADO DE MASCOTAS CON TURNO PARA LA FECHA INGRESADA**

**=============================================================**

**------------------------------------------**

**NUMERO DE TURNO #1**

**------------------------------------------**

**MATRICULA DEL VETERINARIO: 123**

**FECHA -> 13 / 12 / 2020**

**DNI DUEÑO: 24858638**

**Presione una tecla para continuar . . .**

Al ingresar a la opción 3 se muestra en pantalla lo siguiente:

(datos ficticios)

INGRESE LA FECHA DEL TURNO A REGISTRAR.

DIA: 13

MES: 12

AÑO: 2020

INGRESE EL DNI DEL DUEÑO: 44123567

------------------------------------

HISTORIA CLINICA DE LA MASCOTA

------------------------------------

NOMBRE Y APELLIDO: Manuelita Mitre

DOMICILIO: Bernardino Rivadavia 560

DNI DEL DUEÑO: 44123567

LOCALIDAD: San Miguel de Tucuman

FECHA NACIMIENTO: 12/ 2 /2017

PESO MASCOTA: 4,00 kg

EDAD DE LA MASCOTA: 3 AÑOS

TELEFONO DE CONTACTO: 42453523

------------------------------------

REGISTRAR EVOLUCIÓN

------------------------------------

INGRESA LOS DETALLES DE LA ATENCION: aqui debe ingresarse la evolucion de la mascota

----------------------------------------------------

EVOLUCION GUARDADA

------------------------------------------------------

* 1. **MODULO RECEPCION**

Al inicio de este programa se declararon las librerías pertinentes para el desarrollo del mismo, además se declararon constantes las cuales nos facilitaran el nombramiento de cada uno de los archivos utilizados.

Seguidamente se declararon las diferentes estructuras, una estructura es un tipo de dato compuesto que permite almacenar un conjunto de datos de diferente tipo. Los datos que contiene una estructura pueden ser de tipo simple (caracteres, números enteros o de coma flotante etc.) o a su vez de tipo compuesto (vectores, estructuras, listas, etc.).

En un tercer punto se encuentra la función principal del programa, dentro de la cual se incluye la parte primordial del código. En **main(),** incluimos en un primer lugar la función **setlocale(LC\_ALL,”spanish”** la cual nos permite mostrar por pantalla aquellas palabras que contengan acentos y palabras con “ñ”. En primer lugar, se declaran variables numéricas, las cuales contendrán la cantidad de turnos, veterinarios, mascotas y usuarios.

A continuación, se procede a desplegar el menú de usuario, mostrando las respectivas opciones que el usuario posee.

La función **inicio** es necesaria para validad los nombres de usuarios y contraseñas ingresadas con los datos que posea el archivo binario Usuarios.dat, mediante estructuras repetitivas se recorrerá cada registro buscando coincidencias, cuando encuentre el nombre de usuario y contraseña buscado, finalizará la búsqueda y le permitirá al usuario entrar al sistema.

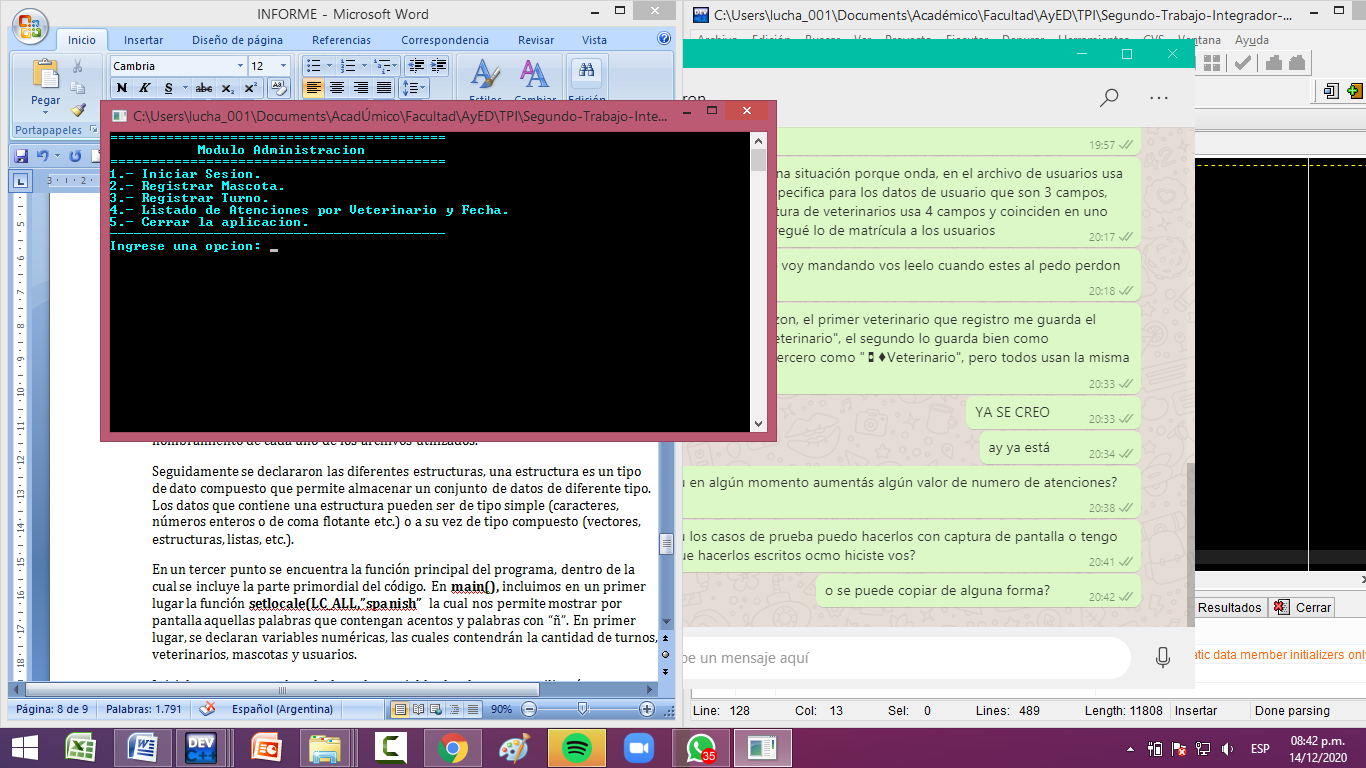
La función **registroM** se encarga de registrar los datos necesarios para la historia clínica de la mascota, estos datos son: nombre y apellido, domicilio, DNI del dueño, localidad, fecha de nacimiento y teléfono. Estos datos son grabados en los campos de un registro llamado Ingreso y copiados en el registro binario Mascotas.dat el cual será utilizado en otros módulos.

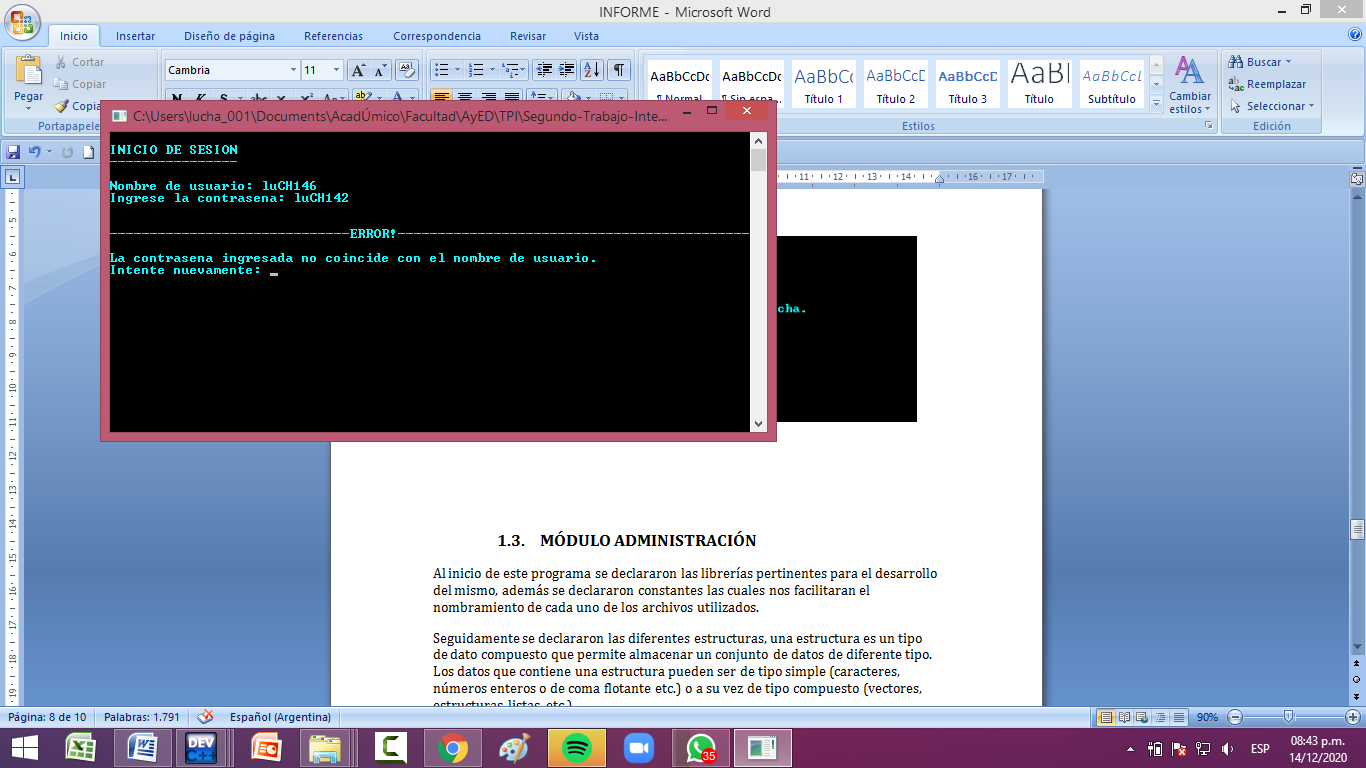
La función **registroT** registra la información perteneciente a un turno solicitado por el dueño de una mascota, la información requerida para dar un turno consta de: matricula del veterinario el cual atenderá a la mascota, DNI del dueño y fecha del turno solicitado, estos datos se guardan en el archivo binario Turnos.dat, este archivo será de utilidad en otros módulos.

La función **listado** toma como referencia el archivo binario Turnos.dat y de acuerdo a los registros que este posea mediante estructuras repetitivas ordenará los turnos de un veterinario determinado de acuerdo a la fecha de atención y al finalizar, mostrará dicho listado.

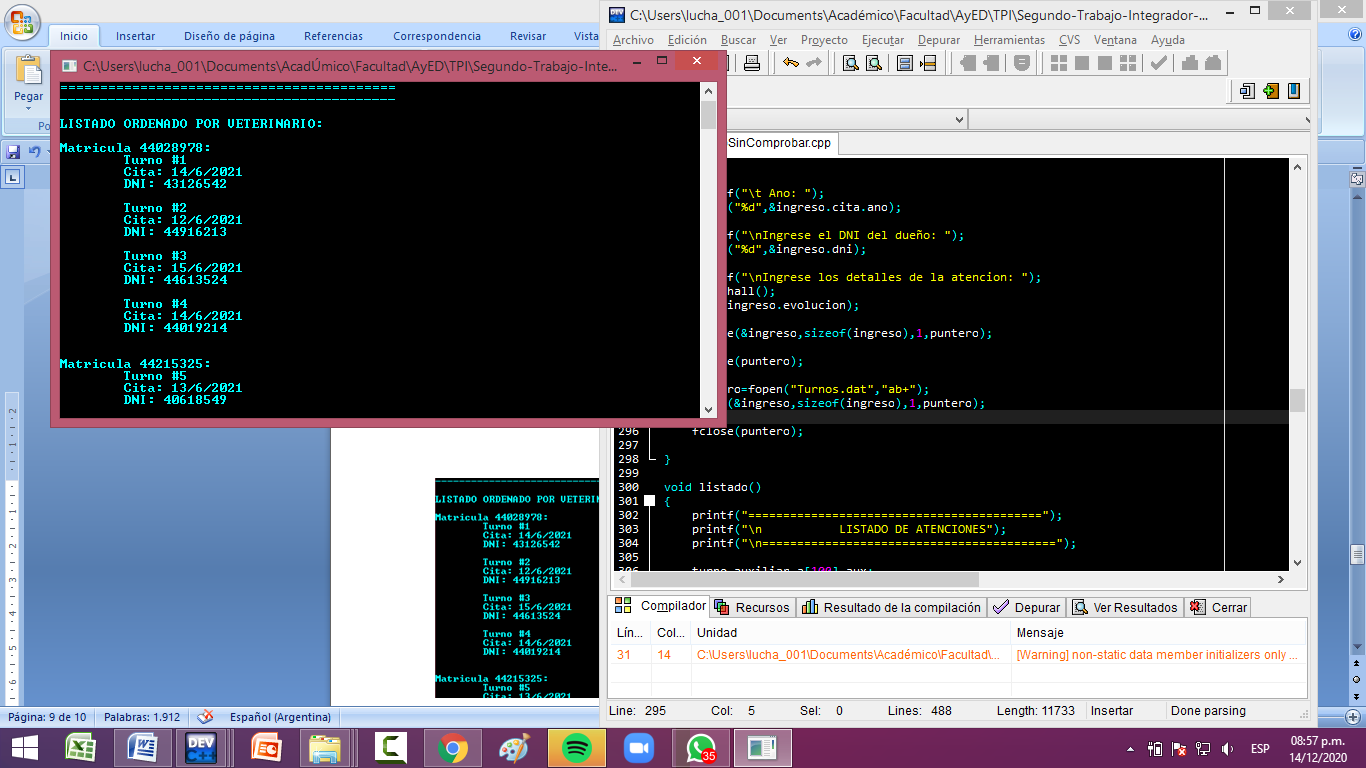
* + 1. **Casos de prueba**

Al iniciar el programa se muestra el siguiente menú:



El programa no permitirá ejecutar las instrucciones correspondientes a la segunda, tercera y cuarta opción si el usuario no ha iniciado sesión antes, por lo tanto, siempre que se inicia el programa, se deberá iniciar sesión para evitar que alguien ajeno a la veterinaria tenga acceso a los datos.  


Una vez que se haya iniciado sesión correctamente, se habilitan las demás opciones del menú.  
Las opciones 2 y 3 funcionan de una manera similar, ambas cumplen la función de solicitar los datos específicos que correspondan, tanto para registrar una nueva mascota, como para registrar un nuevo turno.   
La opción número 4 permite desplegar el listado de atenciones, las cuales se ordenarán por veterinario y se mostrará la fecha de la cita:

****

* 1. **MÓDULO ADMINISTRACIÓN**

Al inicio de este programa se declararon las librerías pertinentes para el desarrollo del mismo, además se declararon constantes las cuales nos facilitaran el nombramiento de cada uno de los archivos utilizados.

Seguidamente se declararon las diferentes estructuras, una estructura es un tipo de dato compuesto que permite almacenar un conjunto de datos de diferente tipo. Los datos que contiene una estructura pueden ser de tipo simple (caracteres, números enteros o de coma flotante etc.) o a su vez de tipo compuesto (vectores, estructuras, listas, etc.).

En un tercer punto se encuentra la función principal del programa, dentro de la cual se incluye la parte primordial del código. En **main(),** incluimos en un primer lugar la función **setlocale(LC\_ALL,”spanish”** la cual nos permite mostrar por pantalla aquellas palabras que contengan acentos y palabras con “ñ”. En primer lugar, se declaran variables numéricas, las cuales contendrán la cantidad de turnos, veterinarios, mascotas y usuarios.

Inicialmente se procede a declarar las variables locales que se utilizarán y dándoles sus valores respectivos para que el código se ejecute de manera correcta y eficaz

A continuación, se procede a desplegar el menú de usuario, mostrando las respectivas opciones que el usuario posee.

La función **listar** extrae del archivo bianario Usuarios.dat los registros pertenecientes a diferentes usuarios, en este caso veterinarios y los muestra por pantalla.

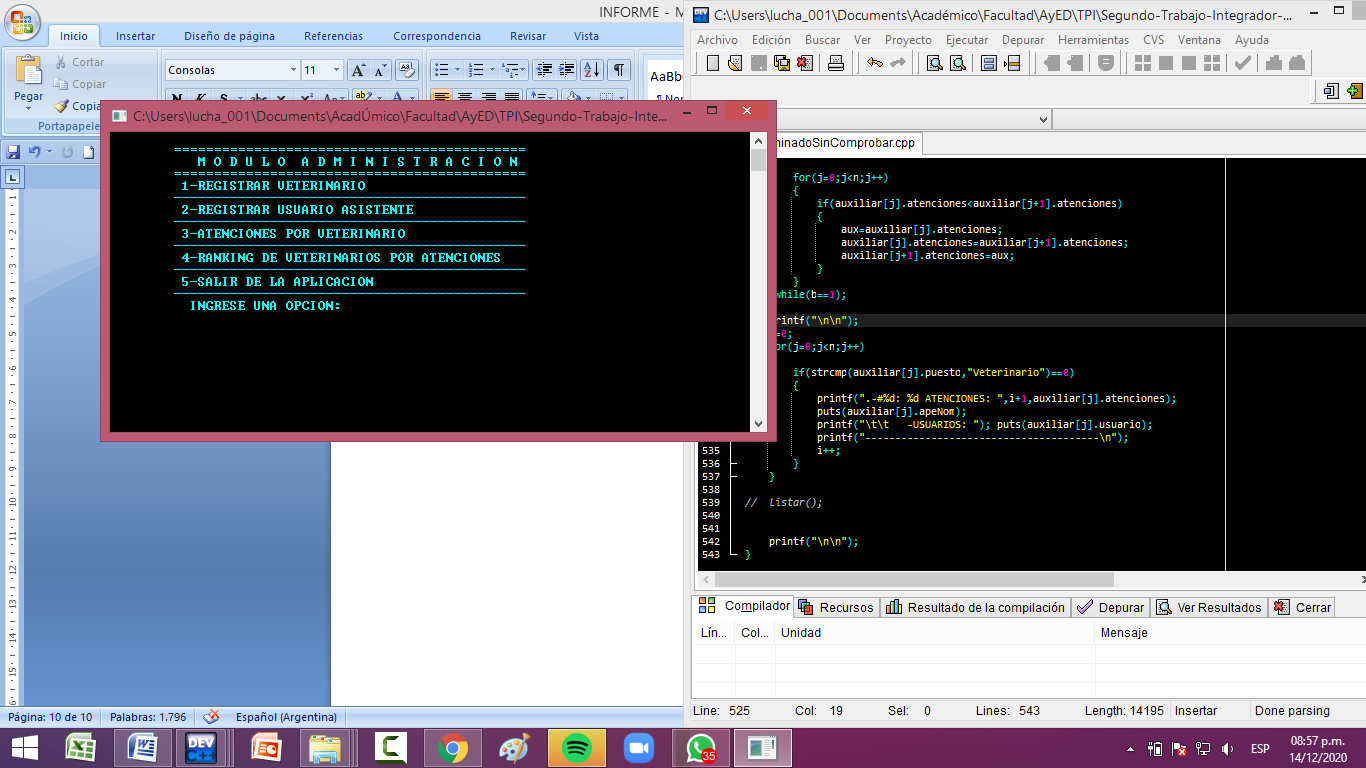
La función **registrar** recibe como parámetros la opción enviada desde el menú principal, este valor será útil para comprobar si la persona que se encuentra utilizando el sistema desea registrar un veterinario o un nuevo asistente, de acuerdo con este valor se procederá a realizar las estructuras correspondientes solicitando el nombre de usuario como así también la contraseña del mismo. Cabe destacar que el nombre de usuario y la contraseña se someterán a un estudio en donde debe cumplir con las condiciones establecidas por el autor de este proyecto, si todo el proceso es favorable, se procede a registrarlos en un archivo binario.

La función **atencionesVet** mostrará de acuerdo a los turnos ingresados, la cantidad de turnos pertenecientes a un veterinario en particular.

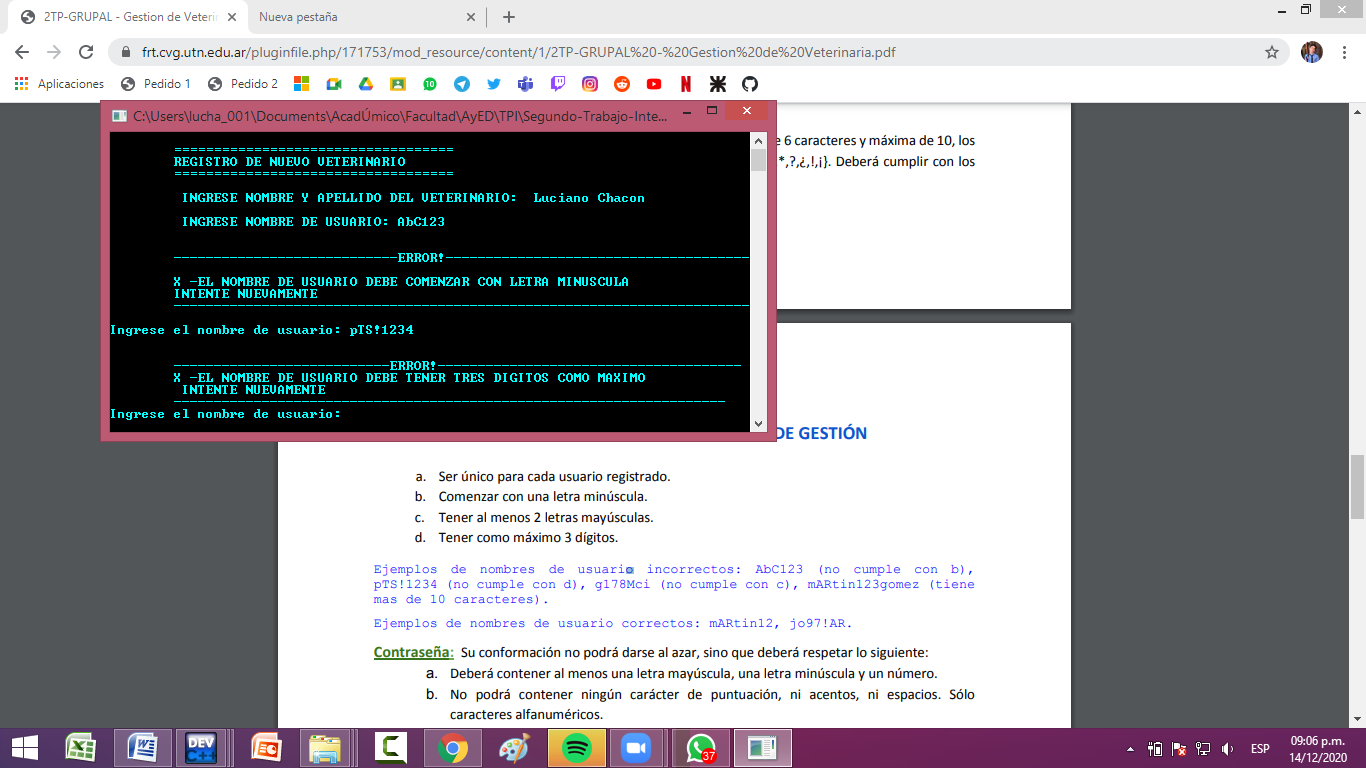
La función **rankingVet** muestra un listado de los veterinarios con mayor número de atenciones, este listado es necesario ya que se le otorgará un bono al veterinario en el puesto numero 1.

* + 1. **Casos de prueba**

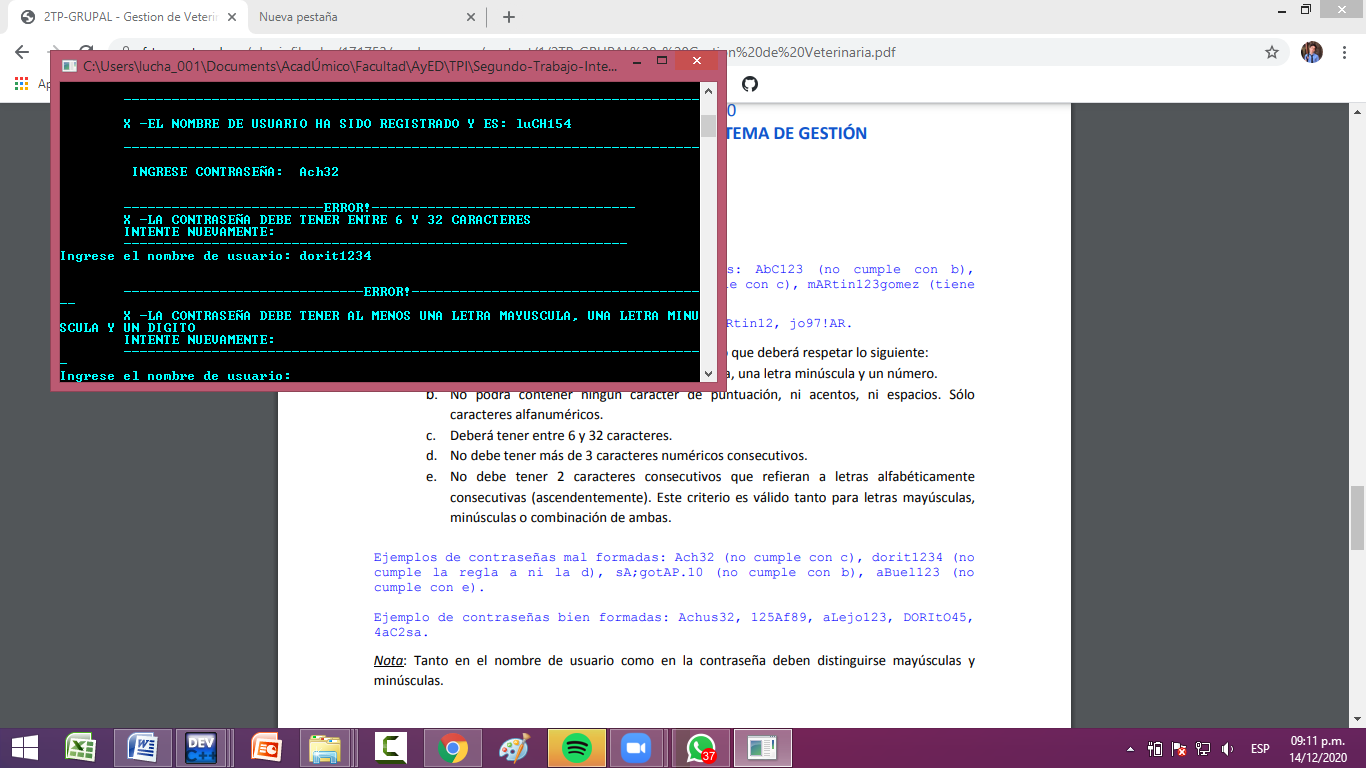
Al iniciar el programa, se mostrará el siguiente menú:



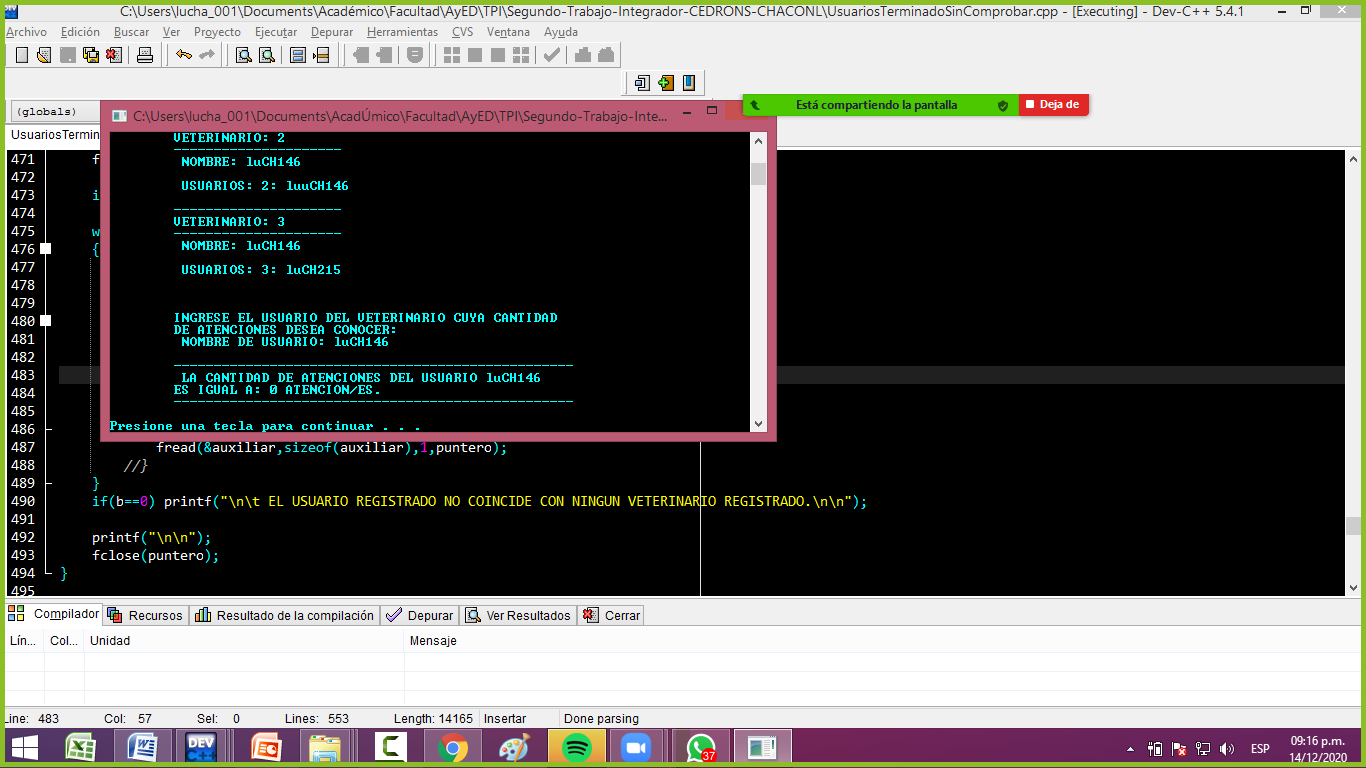
A la hora de registrar tanto un veterinario como un asistente, se solicita en primer lugar el nombre y el apellido de la persona que se desea registrar. A continuación, se solicita un nombre de usuario que debe cumplir con ciertas condiciones para ser registrado, las cuales aparecerán en pantalla en caso de que no se cumplan.



Lo mismo sucede a la hora de ingresar la contraseña:



En caso de elegir la opción numero 3, se mostrará una lista de usuarios y a continuación se solicitará que se ingrese uno de ellos para mostrar el número de atenciones.



En caso de elegir la opción 4, se mostrará una lista de los turnos, ordenados por matrícula del veterinario a realizarla, junto con la fecha de la cita así como también el DNI del dueño de la mascota.

