|  |  |  |
| --- | --- | --- |
| escudofi_color_m2008_jpg | **Carátula para entrega de prácticas** | |
| Facultad de Ingeniería | | Laboratorios de docencia |

Laboratorio de computación . Salas A Y B

|  |  |
| --- | --- |
| Profesor: | Claudia Rodríguez Espinoza |
| Asignatura: | Fundamentos de programación |
| Grupo: | 1104 |
| No de Práctica(s): | N° 13 |
| Integrante(s): | Romero Bernal Rocío Fabiola |
|  |  |
|  |  |
|  |  |
| No. de Equipo de cómputo empleado: | N° 42 |
| Semestre: | 2019-1 |
| Fecha de entrega: | 12/10/18 |
| Observaciones: |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

PRACTICA #13: Lectura y escritura de datos.

Objetivo: Elaborar programas en lenguaje C que requieran el uso de archivos de texto plano en la resolución de problemas, entendiendo a los archivos como un elemento de almacenamiento secundario.

A lo largo del semestre, hemos creado programas capaces de llevar a cabo algunas tareas sencillas mediante el seguimiento de una secuencia lógica, ciclos repetitivos o estructuras de condición, que involucran el uso de determinadas funciones como lo son for,while,switch e if, u otras más complejas que nos permiten trabajar con diferentes valores bajo el nombre de una sola variable, como lo son los arreglos y las cadenas de caracteres las cuales, tal como su nombre lo indica, trabajan con caracteres en lugar de hacerlo con cifras numéricas, permitiéndonos crear programas que recopilen datos del usuario como su nombre, apellido,rfc,etc., abriendo un nuevo camino en nuestra gama de posibilidades para crear un programa; y después llegaron los apuntadores, con los cuales, aprendimos a trabajar a nivel memoria, cuyo uso era importante, pues nos facilitaba en gran medida el uso de las demás funciones, principalmente el de los arreglos y las cadenas de caracteres.

Además entramos en un nuevo nivel de la forma de estructurar un programa con el uso de las funciones secundarias, ya que a medida que añadíamos más funciones dentro de la sintaxis de un programa, este se volvía cada vez más confuso y difícil de comprender, sin embargo, gracias a estas funciones, tuvimos la alternativa de separar las partes de un programa entre la función principal y las demás funciones secundarias que pertenecieran al programa con el único cuidado del tipo de variables(locales o globales), que se estuvieran utilizando, y de esta manera la comprensión del mismo se volvía relativamente sencilla sin perder de vista la tarea con la que tenía que cumplir el programa.

Ahora ya es posible decir que la única limitante para programar a nivel memoria seria nuestra capacidad para plantear estructuras funcionales, que nos permitan cumplir con nuestro objetivo, pues por parte de la teoría, ya nos encontramos lo bastante avanzados para impedir que un obstáculo nos pueda detener; sin embargo, cuando hablamos de trabajar con archivos externos mediante la escritura y lectura de datos, nuestro esquema anterior se encuentra limitado por ser precisamente para trabajar a nivel memoria y no con otro tipo de archivos, aunque no es completamente obsoleto para impedirnos usar las antiguas estructuras aprendidas en esta nueva forma de programar.

Cuando nosotros hablamos de la programación con archivos externos, nos referimos al registro sistemático de datos de un mismo tipo en un archivo respectivamente, para su posterior uso. Y para lograrlo usaremos un apuntador a un archivo, para unificar el sistema de entrada y salida de datos con el buffer por donde se transportaran los datos.

Este es típicamente el primer paso para comenzar a trabajar a nivel archivo; ya que posteriormente intentaremos abrir, crear, añadir texto o leer el contenido de un archivo externo con el uso de los modos siguientes:

r: Abre un archivo de texto para lectura.

w: Crea un archivo de texto para escritura.

a: Abre un archivo de texto para añadir.

r+: Abre un archivo de texto para lectura / escritura.

w+: Crea un archivo de texto para lectura / escritura.

a+: Añade o crea un archivo de texto para lectura / escritura.

rb: Abre un archivo en modo lectura y binario.

wb: Crea un archivo en modo escritura y binario.

Por otra parte las funciones Fopen, Fclose, Fgets, Fputs, Fprintf, nos permiten abrir, cerrar, leer o añadir secuencias (de cualquier tipo de variables que se permita en C++) al archivo con el cual estemos trabajando.

ACTIVIDAD 1: “Agenda”.

El siguiente programa nos permite introducir los datos de tres usuarios distintos (su nombre, apellidos, dirección, y número de teléfono) cuyos datos son guardados en un archivo externo bajo el nombre de agenda. Aquí hacemos uso de una estructura que nos permite guardar variables de diferente tipo bajo el nombre de la estructura datos.

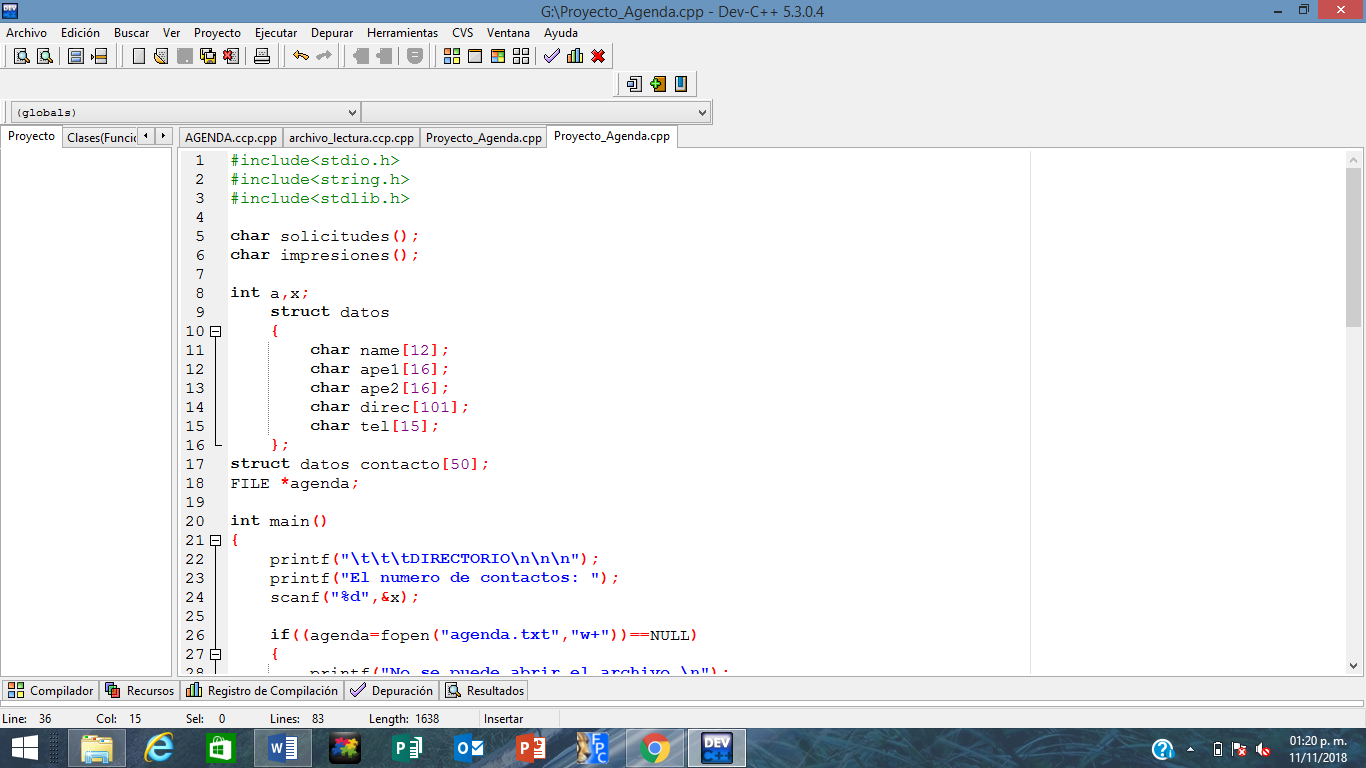
Su sintaxis es la siguiente:

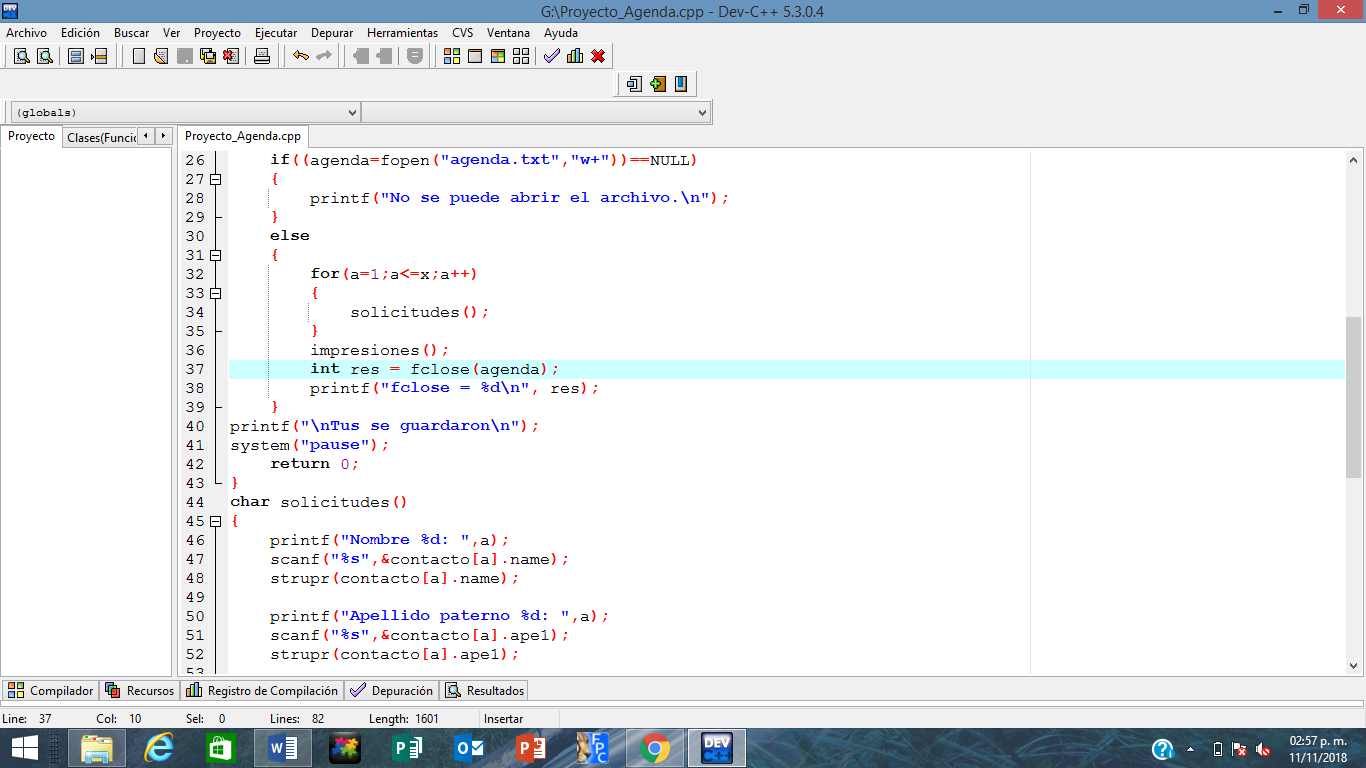
Este programa, además de la función principal, utiliza las funciones secundarias char solicitudes, char impresiones para:

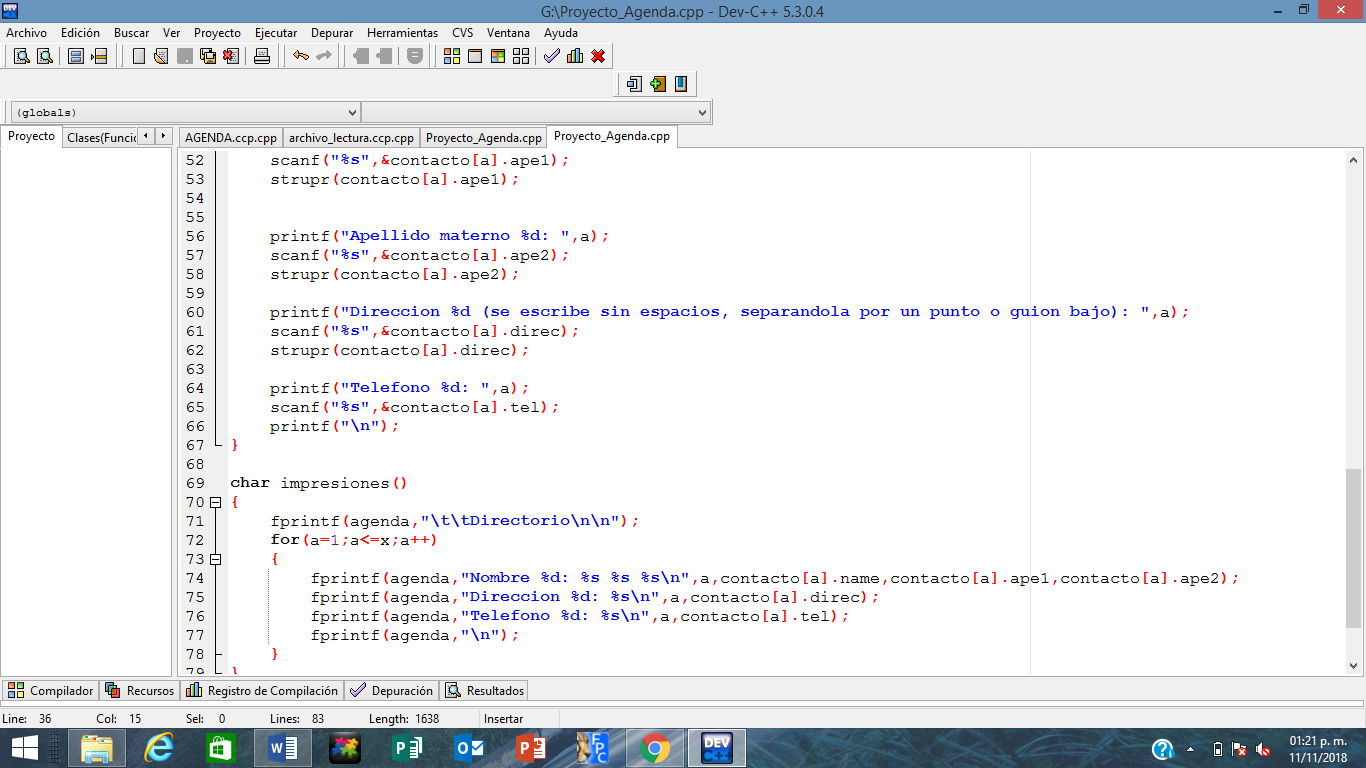
Int main: Es la función principal, en ella vemos que se pide el número de contactos que se desea registrar y la creación de un archivo que lleva el nombre de agenda. Además con fopen se abre una secuencia que posteriormente añadiremos al archivo, con un condicionador if, que nos informa si el archivo existe y en caso de que si exista, le permite al buffer mandar los datos correspondientes al archivo. Y, finalmente encontramos a las llamadas a las funciones solicitudes e impresiones y la función fclose la cual cierra la secuencia anteriormente abierta.

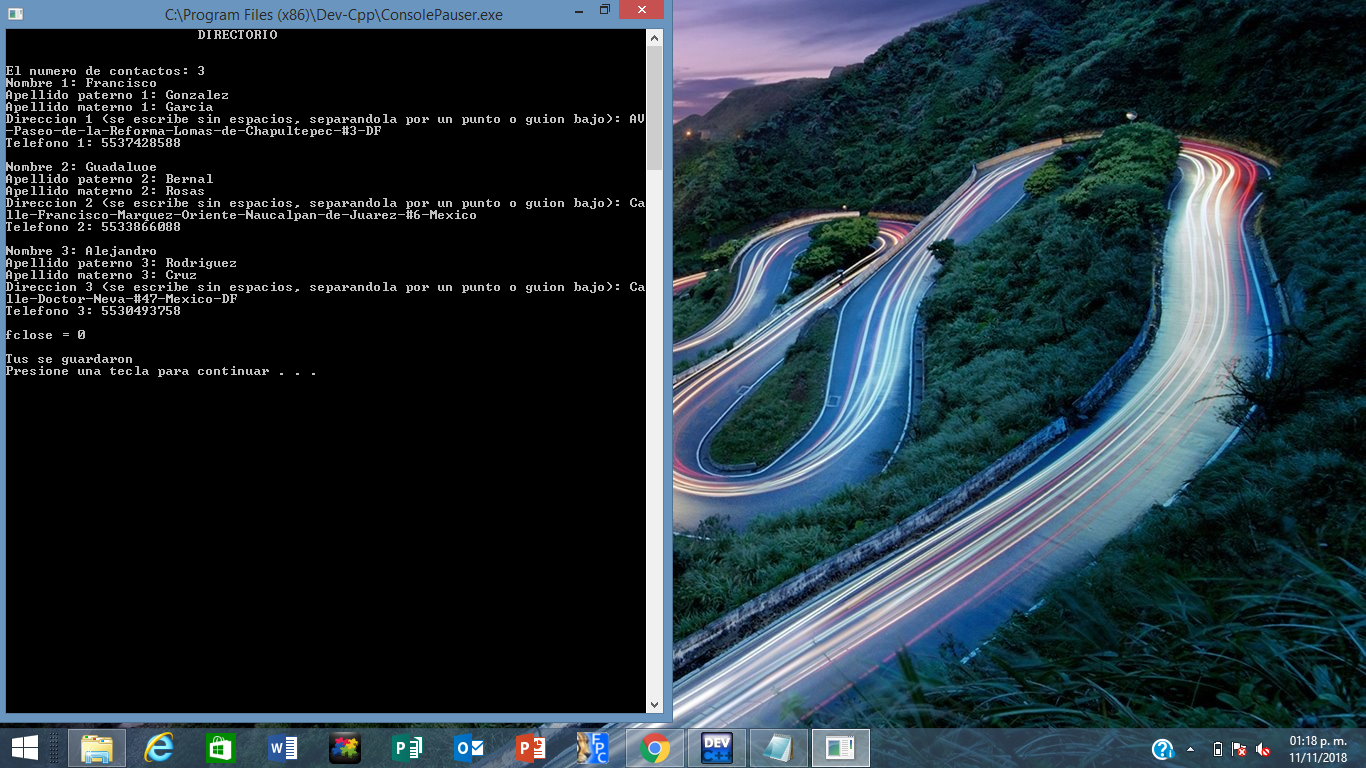
Char solicitudes: Pide la información del número de contactos que se le indique pedir al programa en la función principal, con ayuda de una función for (note que las variables son globales y por ello no deben especificarse las variables con las que se está trabajando en cada función).

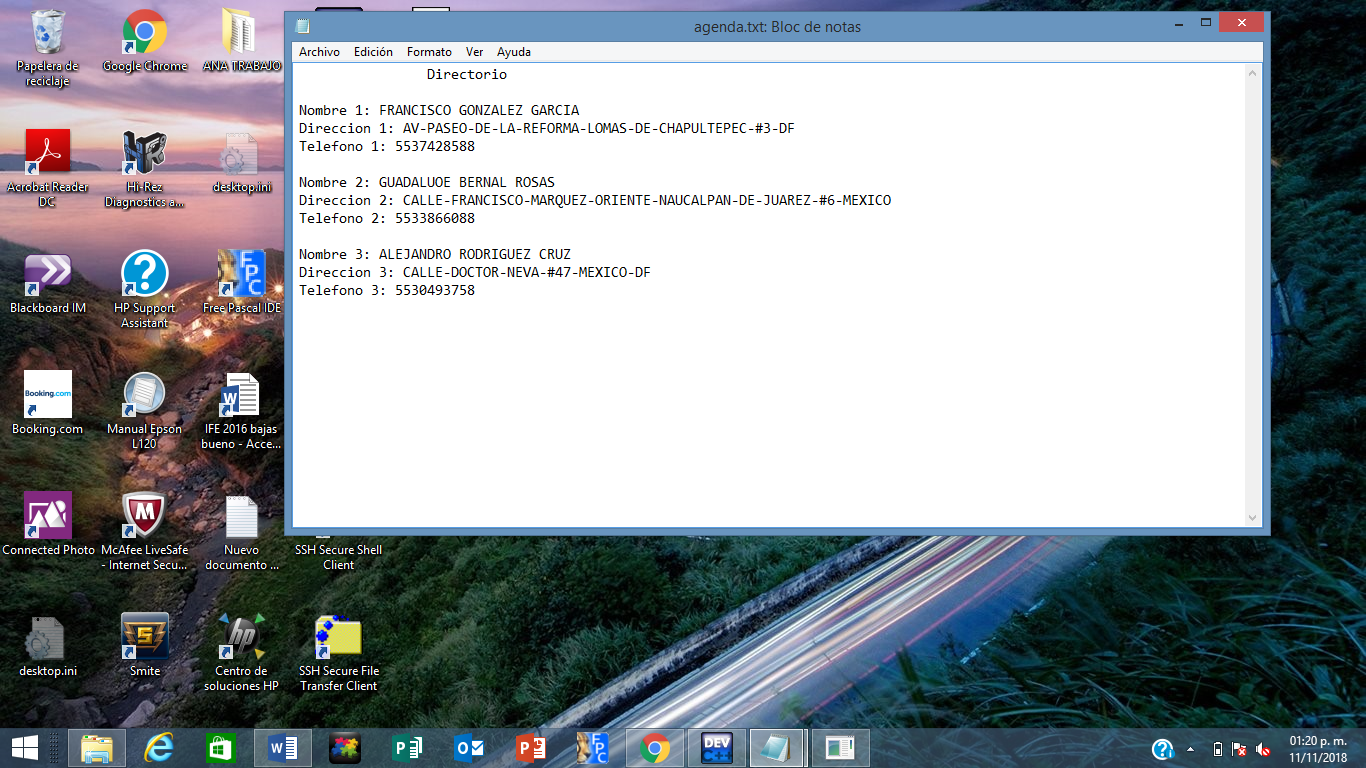
Char impresiones: imprime los datos obtenidos de los contactos en el archivo externo. Aquí encontramos una función fprintf que se encarga de imprimir los datos recopilados en el archivo agenda.











CONCLUSIONES: Es sumamente importante que cuando estemos trabajando con estructuras y lectura y escritura de datos de un archivo externo, procuremos añadir por separado las funciones a cada uno de los datos que pertenecen a la estructura bajo la declaración Nombre estructura. Nombre del miembro, pues en caso de solo añadir el nombre de la estructura, el programa nos mandara un mensaje de error y será imposible ejecutar el programa. Más aun cuando se trate de arreglos de estructuras, donde además tendremos que señalar el tamaño del arreglo de estructura o la variable que se le asigno para trabajar con ciclos repetitivos; y cualquiera de las dos opciones deberá ir entre corchetes [].