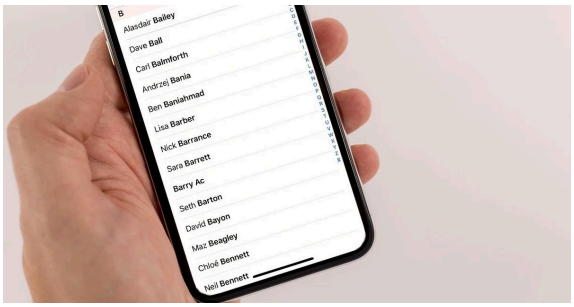
Universidad Francisco de Vitoria

UFV Madrid



Universidad Francisco de Vitoria

Práctica
Agenda telefónica

Introducción a la programación

PRÁCTICA 2: AGENDA TELEFÓNICA

El objetivo de esta práctica es comprobar que el alumno ha comprendido y utiliza correctamente los arrays y las estructuras, así como las estructuras básicas del lenguaje (condicionales, bucles, declaración de variables, …).

Especificación de la agenda

En esta práctica se trata de implementar una agenda telefónica con las operaciones básicas CRUD (Create, Read, Update, Delete) para el manejo de la misma.

El elemento básico de la agenda la siguiente estructura:

```
#define MAX_CAD 100

struct persona{
    char nombre[MAX_CAD];
    char apellidos[MAX_CAD];
    unsigned long telefono;
    char email[MAX_CAD];
};
```

La estructura persona tiene el nombre y los apellidos de contacto que se quiere almacenar en la agenda, así como el numero de teléfono y la dirección de email.

Para considerar que un contacto está vacío se va a utilizar la siguiente convención: todos los campos de texto deben almacenar la cadena "?" y los campos numéricos deben almacenar un 0.

La operaciones a realizar sobre la agenda son las siguientes (estas son las funciones y procedimiento que se debe implementar de forma obligatoria y son genéricas):

```
int insertarPersona(struct persona agenda[], int capacidad,
                    struct persona nuevo);
```

Esta función permite añadir un nuevo contacto (persona) a la *agenda*, que se pasa en el parámetro *nuevo*, siempre que no se haya alcanzado la *capacidad* máxima de la misma.

El nuevo contacto se añade en la primera posición libre que se encuentre.

La función devuelve 1 si se ha podido insertar el contacto en la agenda, y devuelve 0 en caso que la agenda esté llena y no se pueda almacenar el contacto.

Si el almacenamiento de los contactos se hace forma ordenada por apellido se obtendrá una puntuación superior en este apartado.

```
void listarAgenda(struct persona agenda[], int capacidad);
```

Esta función imprime en pantalla todos los contactos (no vacios) de la agenda acorde al siguiente formato (tal cual están almacenados - respeta mayúsculas y minúsculas)

```
#orden          Apellido, Nombre      telefono   email
#orden          Apellido, Nombre      telefono   email
#orden          Apellido, Nombre      telefono   email
```

A continuación se muestra un ejemplo:

```
1 Sanchez, Luis 912345678       luis.sanchez@luis.com
2 Lopez, Emilio 918765432       emilio@emilio.es
```

En caso que la agenda no tenga contactos almacenados, se debe mostrar en pantalla el mensaje “Agenda vacia”.

```
struct persona buscarPersona(struct persona agenda[], int
                                    capacidad, char nombre[]);
```

Esta función busca en la *agenda* un contacto con el mismo nombre que el que se pasa en el parámetro *nombre* (NO DISTINGUE ENTRE MAYÚSCULAS Y MINÚSCULAS, el nombre Paco es igual que paco). En caso de encontrarlo, devuelve la información del contacto en un nuevo contacto con la información encontrada en la agenda.

En caso de tener más de un contacto con el mismo nombre, se debe devolver la primera ocurrencia encontrada. Si no hay ningún contacto con *nombre*, se devolverá un contacto vacío (campos de texto con ‘?’ y campos numéricos con valor 0).

```
int borrarPersona(struct persona agenda[], int capacidad, char
                        nombre[]);
```

Esta función borra de la *agenda* todos los contactos con el mismo nombre que el que se pasa en el parámetro *nombre* (NO DISTINGUE ENTRE MAYÚSCULAS Y MINÚSCULAS, el nombre Paco es igual que paco), y devuelve el número de contacto borrados.

El borrado se puede hacer de dos formas:

- Borrado no compacto: se elimina el contacto poniendo, haciendo que el contacto sea vacío (campos de texto con ‘?’ y campos numéricos con valor 0). Este tipo de borrado deja huecos en la agenda.
- Borrado compacto: elimina el contacto y el hueco dejado en la agenda por el borrado del mismo. Si se realiza este tipo de borrado se obtendrá una puntuación superior en este apartado.

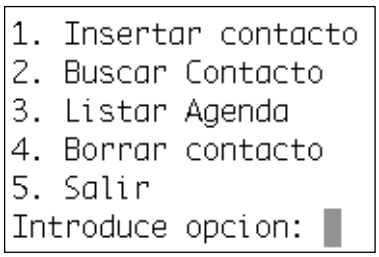
Funcionamiento de la agenda

Asociado a las funciones anteriores (FUNCIONES GENÉRICAS Y QUE DEBEN FUNCIONAR APARA CUALQUIER TAMAÑO DE AGENDA) hay que implementar el programa principal para la gestión de la agenda, acorde al siguiente menú de usuario.

Para definir la agenda, vamos a suponer que tenemos un máximo de 100 persona en la misma (pero podría ser cualquier valor). Si se alcanza el numero máximo de contactos en la agenda, no se podrán almacenar más contactos a partir de ese momento, mientras no se libere espacio. Cuando se inicia la agenda, todos los contactos deben tener en los campos de texto el símbolo ‘?’ y en los campos numéricos un 0 (esto también debe ocurrir cuando se elimina un contacto de la agenda)

PRÁCTICA 2: AGENDA TELEFÓNICA

Si se almacena la agenda en un fichero de texto, antes de comenzar con el programa hay que cargar desde el fichero todos los contactos almacenados. Si el fichero no existe, se debe crear uno nuevo con el nombre agenda.txt y mostrar el menú de usuario. Si no se puede cargar el fichero, hay que mostrar el mensaje "ERROR: imposible inicializar la agenda desde fichero" y terminar el programa. El fichero debe guardar cada ítem en una línea diferente, separando por ; cada uno de los campos (nombre;apellidos;telefono;email)



El menú de la aplicación debe contemplar los diversos errores que pueda cometer el usuario y evitarlos (por ejemplo, introducir la opción 6).

El funcionamiento de cada una de las opciones es el que se describe a continuación

Insertar contacto

El programa debe pedir los datos del contacto que se quiere introducir por teclado, el usuario los teclea (se almacenan tal cual los teclea el usuario), si es posible almacenar el contacto se muestra el mensaje "Contacto almacenado con exito", y si no es posible el programa debe mostrar el mensaje "Error, el contacto no se ha almacenado".

Cuando se termina la operación se debe volver al menú principal.

Buscar contacto

El programa debe pedir el nombre del contacto a buscar y el usuario lo introduce por teclado. Si el contacto no se encuentra en la agenda se muestra el mensaje "el contacto no existe". En otro caso, se muestra la información del contacto en el siguiente formato:

```
Apellido, Nombre      telefono   email
```

Listar agenda

El programa debe mostrar por pantalla todos los contactos almacenados en la agenda acorde a lo especificado en la función *listarAgenda*.

Borrar contacto

El programa debe pedir el nombre del contacto a borrar y el usuario lo introduce por teclado. El programa elimina de la agenda los contactos acorde a los especificado en la función *borrarPersona* e imprimir en pantalla el siguiente mensaje "X contacto(s) eliminado(s)" siendo X el numero de contactos eliminados en la función *borrarPersona*.

Salir

El programa debe terminar. *Si se almacena la agenda en un fichero de texto (agenda.txt) se debe actualizar el fichero con todas las modificaciones realizadas en la misma. Si el guardado del fichero se realiza de forma correcta se debe mostrar el mensaje "Agenda guardada correctamente" y en caso contrario se mostrará el mensaje "ERROR: Agenda no guardada"*

PRÁCTICA 2: AGENDA TELEFÓNICA

Entrega

Esta práctica es de realización obligatoria para todos los alumnos. **Se puede realizar de forma individual o en grupos de 2 alumnos**.

Plazo de entrega: **23:59 horas del día 21 de abril de 2020**.

Habrá que entregar a través de Moodle un **practica2.c** con el siguiente contenido:

- Un programa principal que contenga el menú para la gestión de la mochila.
- Cada una de las funciones descritas en el enunciado, acorde a los nombres dados en el mismo.
- Funciones adicionales que se utilicen para mejorar la estructura del código
- Si alguna de las partes no se implementa, el programa principal debe indicarlo en pantalla con el siguiente mensaje: `"Parte no realizada"`

La práctica SÓLO debe entregarla uno de los alumnos del grupo (el que aparezca en primer lugar en los nombres comentados)

Se considerará **plagio** entregar un código idéntico total o parcialmente al de otro compañero. Tampoco se admitirá un código idéntico a otro presente en cualquier página de Internet, incluso aun citando la fuente. **El profesor dispone y usará herramientas de detección automática de plagio.**

PRÁCTICA 2: AGENDA TELEFÓNICA

Puntuación

Estos ejercicios obligatorios se puntuará de 0 a 10, nota que se contabilizará en el 20% de la nota final de la asignatura (prácticas) en el caso de la evaluación continua. Las prácticas no entregadas se computarán como un 0.

Elementos a evaluar:

- **Incluir el nombre de los autores como las dos primeras líneas (comentadas) del fichero practica2.c**. Si para la correcta compilación del programa hay que incluir algún parámetro opcional al compilador, se debe incluir en una comentario debajo del nombre de alumno la correcta llamada al compilador.
- La no existencia de errores sintácticos en el código (es decir, **el código debe compilar**). En caso contrario, **una práctica cuyo código fuente dé errores de compilación se considerará suspensa directamente**.
- Se valorará positivamente que no haya errores lógicos.
- Limpieza y claridad en el código.
- Presencia de comentarios en el código.
- Errores descritos en el documento "*Errores a evitar.pdf*" que se encuentra en Moodle junto con el enunciado de la práctica.

De forma numérica los pesos de cada parte de la practica son

insertarPersona	1,5
mantener la agenda ordenada por apellido de persona	1,5
listarAgenda	1
buscarPersona	1,5
borrarPersona	1,5
mantener la agenda sin huecos (agenda compacta)	1,5
Programa principal	1,5
[EXTRA y OPCIONAL] almacenamiento en fichero de texto (lectura/escritura)	3
Total	10

La puntuación máxima a obtener puede ser 13 puntos si se realizan todos los apartados. **La parte "EXTRA y OPCIONAL" sólo se considerará si se han realizado TODOS los apartados obligatorios de la práctica.** Para la realización de la parte "EXTRA y OPCIONAL" el alumno debe buscar el material necesario para su realización, puesto que el tema de ficheros no se ha visto en las clases de teoría.