

## Práctica #3

### Base de datos estudiantes.

#### Objetivo

Implementar dos programa para salvar información de estudiantes en un fichero como si fuera una especie de base de datos. El programa debe permitir agregar estudiantes, imprimir el listados de los estudiantes presentes y borrar la base de datos. El programa debe almacenar los datos en modo binario. El programa debe poder compilarse utilizando make. A continuación los requerimientos:

1. El nombre del programa debe ser **studentdb**
2. Los datos serán almacenados en un archivo sin extension llamado studentdbData
3. Al iniciar la aplicación, si el archivo donde se almacenará la información no existe entonces debe crearlo.
4. La aplicación debe asignar a cada estudiante un id que debe ser un entero que se incremente al agregar un estudiante. El id del primer estudiante en un fichero vacío debe ser uno.
5. Todas las funciones desarrolladas para interactuar con la base de datos deben de estar un fichero de nombre studentdb.c
6. El archivo principal del programa debe ser main.c, en este debe manejarse todo lo asociado al menu.

#### Formato de entrega

Todos los archivos \*.c del programa deben estar en un solo directorio. En este directorio debe haber un Makefile capaz de compilar el programa al ejecutar **Make all**. El Makefile también debe permitir eliminar el programas al ejecutar **Make clean**.

Deben entregar un folder que contenga todos los archivos. No deben haber subfolders, solo un folder. El folder debe tener el formato prac\_{número de practica}\_{primer apellido}\_{primer nombre}\_{matricula}. Al momento de entregar deben hacer un zip file con el mismo formato prac\_{número de practica}\_{primer apellido}\_{primer nombre}\_{matricula}. **Los símbolos de { } no deben ir en los nombres, es solo para encerrar el template de como sería el formato.**

#### Descripción del menu.

Al iniciar el programa debe presentar un menú con las siguientes opciones.

1. Agregar estudiante
2. Imprimir estudiantes

3. Buscar por id
4. Borrar base de datos
5. Salir

Introduzca selección:

**(1 .Agregar estudiante)** Esta opción debe pedir la siguiente información al usuario en el siguiente orden.

1. Primer Nombre:
2. Primer Apellido:
3. Edad:
4. GPA:

El usuario debe presionar ENTER al momento de introducir cada elemento. Al introducir el GPA debe volver al menú principal. No debe imprimir más nada luego de introducir el GPA. El Nombre y Apellido pueden tener 50 caracteres máximo incluyendo el sentinela '\0'. Si se introduce mas de 50 el resto sera descartado.. El GPA es entre 0 y 4, con dos dígitos después del punto.

**(2. Imprimir estudiantes)** Debe imprimir el listado de estudiantes en pantalla con el siguiente formato. {id} - {primer nombre} {primer apellido} {edad} {gpa con formato 0.00, dos dígitos después del punto}. Al finalizar de imprimir la lista debe volver a imprimir el menú principal.

Ejemplo:

- 1 – Ricardo Martinez 29 3.5
- 2 – Fernando Vargas 27 3.0

No deben usar tabulaciones, deben separar cada elemento con un espacio en blanco.

**(3. Buscar por id)** Debe pedirle al usuario introducir un id y luego imprimir el dato del estudiante utilizando el formato de la lista. Luego de impreso debe imprimir el menú principal.

Ejemplo.

Introduzca el id: 2

- 2 – Ferando Vargas 27 3.0

**(4. Borrar base de datos)** Solo debe imprimir el mensaje - base de datos borrada -. Esto debe borrar el archivo de base de datos creado.

// Ignorar esto

El programa debe permitir recibir los argumentos del menu principal directamente desde la consola. En dicho caso al terminar la función debe cerrarse.

Ejemplo programa dbA -o1