

Universidad Simón Bolívar

Departamento de Computación y T.I.

Taller de Sistemas de Operación I

Abril-Junio 2015

Tarea 1: Programación en Lenguaje C

5 pts

Objetivos

El objetivo principal de la tarea es que el estudiante adquiera destrezas en el lenguaje de programación C. Se hará énfasis en los siguientes elementos: estructuras de datos dinámicas, manejo de apuntadores y operaciones de E/S.

Enunciado

Se desea realizar un programa para actualizar un conjunto de datos que contiene preguntas y respuestas. El programa se desarrollará en lenguaje C y debe permitir operaciones de actualización y consulta sobre las preguntas existentes. A continuación se describen los componentes más importantes del programa.

Las Preguntas

Cada pregunta tiene los siguientes atributos:

- Un código o identificador de la pregunta.
- El área a la que pertenece la pregunta: Historia (H), Geografía (G), Tecnología e Informática (T), Cine (C) y Literatura (L).
- Nivel de complejidad (0=básico, 1= intermedio, 2=avanzado).
- Respuestas: Cada pregunta tiene solamente tres posibles respuestas de las cuales sólo una es la correcta.
- Respuesta correcta: Es un entero del 1 al 3 que indica cuál es la respuesta correcta a la pregunta formulada: la primera (1), la segunda (2) o la tercera (3).

El conjunto de datos donde se almacenan las preguntas y sus atributos es un archivo de texto con el formato que se especificará más adelante.

El programa

Debe ofrecer una interfaz amigable que le permita al usuario seleccionar a través de un menú principal cualquiera de las siguientes opciones:

Leer la base de datos: Cuando el usuario elige esta opción el programa lee del archivo de texto todas las preguntas disponibles junto con sus atributos, colocando la información en una estructura de datos en memoria. La idea es que todas las modificaciones que realiza el usuario mientras interactúa con el programa (durante una sesión) se realicen en la memoria y sólo se escriban en el archivo al final de la sesión o cuando el usuario lo especifique a través de la opción *Salvar la base de datos*. Si se trata de realizar cualquiera de las operaciones que se describirán a continuación (consulta, eliminación, etc.) sin haber leído los datos del archivo, el sistema indicará un error. Por el contrario, si la operación de Leer la base de datos se invoca más de una vez dentro de la misma sesión, el programa indicará al usuario que los datos ya están en la memoria.

Consultar todas las preguntas que están en la base de datos: A través de esta opción se muestran todas las preguntas de la base de datos (incluyendo las que han sido añadidas en la sesión actual) con todos sus atributos: respuestas y respuesta correcta, área, código y nivel de complejidad.

Consultar todas las preguntas de un determinado nivel de complejidad. Dado un nivel de complejidad se muestran todas las preguntas que existen actualmente en ese nivel.

Eliminar una pregunta: Dado el código de una pregunta, la misma se elimina del conjunto de preguntas. Si no existe una pregunta con ese código el programa muestra un error.

Insertar una pregunta: Cuando se selecciona esta opción, se debe mostrar una interfaz al usuario que le permita insertar una pregunta con todos sus atributos incluyendo el código. El programa debe también presentar una opción para que el usuario confirme que ya ha introducido todos los atributos de una nueva pregunta (esta opción puede ser una simple pregunta a la que se responda si/no). El usuario es responsable de introducir todos los atributos de la pregunta correctamente. El programa no validará que falten atributos o que se introduzca una pregunta con el mismo código de otra ya existente.

Salvar la base de datos: Cuando se selecciona esta opción se debe salvar en un archivo de texto toda la información que se tiene en la memoria relacionada con las preguntas. Un usuario puede salvar varias veces la base de datos mientras interactúa con el programa.

Salir: Con esta opción el usuario culmina su sesión. Si no ha salvado la base de datos, el sistema lo hará por él. Al seleccionar esta opción se termina el programa.

Detalles de Implementación

Formato del Archivo de Entrada

El archivo donde se almacenarán las preguntas y sus atributos es de tipo texto y tiene el siguiente formato:

Código nivel área string1 string2 string3 string4 respuesta

Donde:

Código: es un entero que identifica unívocamente la pregunta.

Nivel: es un entero cuyos valores están comprendidos entre el 0 y el 2, y que indica el nivel de complejidad de la pregunta. El nivel 0 se usa para preguntas básicas, el nivel 1 para preguntas de un nivel intermedio y el nivel 2 para preguntas avanzadas.

Área: es un carácter en mayúscula que determina el área de la pregunta. Las posibilidades son: Historia (H), Geografía (G), Tecnología e Informática (T), Cine (C), Literatura (L), y Matemáticas (M).

String1: es la pregunta y va delimitada entre comillas dobles ("").

String2: es la primera respuesta posible y va delimitada entre comillas dobles ("").

String3: es la segunda respuesta posible, y va delimitada entre comillas dobles ("").

String4: es la tercera respuesta posible, y va delimitada entre comillas dobles ("").

respuesta: es un entero del 1 al 3 que indica cuál es la respuesta correcta a la pregunta.

Un ejemplo del contenido del archivo sería el siguiente:

```
1 1 C "Quién es la actriz principal de la película Los Puentes de Madison" "Meryl Streep"
"Penélope Cruz" "Jodie Foster" 1
2 0 L "Quién es el autor de la novela Doña Bárbara" "Miguel O. Silva" "Rómulo
Gallegos" "Joaquín Crespo" 2
13 0 T "Linux es" "Un compilador" "Un navegador" "Un sistema de operación" 3
9 1 G "Cuál es la capital de Alemania" "Munich" "Berlin" "Atenas" 2
20 1 G "Dónde queda la Isla Ibiza" "En el Mediterráneo" "En el Pacífico" "En el Caribe"
1
```

Observaciones: Cuando el programa salva el conjunto de datos en el disco debe eliminar el archivo anterior y crear uno nuevo conservando el nombre y formato del archivo original.

- Cada uno de los campos que conforman una línea del archivo (código, nivel, área, pregunta, primera respuesta, etc) van separados por un espacio en blanco. Se debe respetar el formato del archivo, recuerde que su programa debe funcionar con el archivo de prueba de los profesores del taller.
- Puede suponer que los strings correspondientes a las preguntas y/o respuestas no contienen comillas dobles. Estas sólo se usan para delimitar el comienzo y el fin del string. Por ejemplo, no es posible encontrar una pregunta como:

"Cuál es el significado de la palabra "overhead""

En su lugar, la pregunta estaría en la base de datos de la siguiente manera:

"Cuál es el significado de la palabra overhead"

Estructuras de datos

Dado que no hay un máximo para el número de preguntas que se pueden almacenar en el archivo que simula la base de datos, la estructura de datos que albergue en la memoria a estas preguntas debe ser dinámica. Ud. debe implementar listas enlazadas para almacenar las preguntas. ([Ejemplo de Listas Enlazadas](#))

Puede suponer además que el número total de caracteres de las preguntas y respuestas no sobrepasará los 100 caracteres.

Invocación del programa

El programa se invocará de la siguiente forma:

\$ *preguntas archivo*

donde:

- *preguntas* es el nombre del programa
- *archivo*: es el nombre del archivo donde se encuentran las preguntas y sus atributos.

Sobre la Entrega

La entrega se debe realizar el lunes de la semana 4 antes de las 11:59pm, a través del Aula Virtual para lo cual deberá crear el directorio Tarea dentro de la carpeta de documentos de su grupo. Note que debe estar suscrito a algún grupo en Aula Virtual para poder optar a esta opción, no espere al día de la entrega para notificar que tiene problemas o que no se ha registrado. En ese directorio debe colocar un archivo tar.gz que contenga únicamente los fuentes (archivos .c y .h) de su proyecto y un archivo Makefile para generar los ejecutables. Proyecto que no pueda compilarse usando las reglas dadas en el archivo Makefile, no será corregido.

Notas adicionales

- El programa debe poder compilarse y ejecutarse en cualquiera de las estaciones (Linux) del LDC. Si Ud. realizó el programa en su casa o en la USB, pero en alguna otra plataforma hardware-software, debe asegurarse antes de la entrega que su proyecto funcione en las estaciones Linux antes mencionadas.
- El programa debe hacer uso eficiente de los recursos que administra el sistema operación: memoria, espacio en disco, etc.
- Debe respetar las especificaciones que se le dan en el enunciado.