



Departamento de Lenguajes
y Ciencias de la Computación

LABORATORIO DE PROGRAMACIÓN

Examen Ordinario de Septiembre

1º Gestión Curso 2008/2009

11/09/2009 a las 9:30

APELLIDOS _____ NOMBRE _____

DNI _____ ORDENADOR _____ GRUPO (A/B) _____

La Federación Internacional de Baloncesto (FIBA) con motivo del Eurobasket que se va a celebrar este año en Polonia, nos ha pedido que desarrollemos una aplicación llamada `fiba.cpp` para almacenar los equipos inscritos junto con su ranking. Para ello usaremos un árbol binario de búsqueda cuya clave de inserción/búsqueda será el nombre del equipo. El programa deberá mostrar el siguiente menú de opciones:

Nombre: (Apellidos, Nombre)	Curso: 1º
Especialidad: Gestión	Grupo: A/B
Puesto: número de ordenador	Fecha: 11/09/2009

Eurobasket 2009
=====

- A. Insertar Equipo.
- B. Listar por Nombre.
- C. Salvar Equipos a Fichero.
- D. Borrar Equipo por Nombre.
- E. Cargar Equipos desde Fichero.
- F. Listar por Ranking.
- G. Borrar Equipo por Ranking.
- X. Salir del Programa

Introduzca Opción:

A. **Insertar Equipo.** Se pedirá desde teclado el nombre y ranking del equipo y se insertará en el árbol. Si ya existiera un equipo con el mismo nombre o no hubiera memoria se informaría del error.

B. **Listar por Nombre.** Se mostrará por pantalla un listado de todos los equipos ordenados alfabéticamente.

C. **Salvar Equipos a Fichero.** Se pedirá el nombre de un fichero de texto y se salvará toda la información con el formato adjunto. Si el fichero no existe se creará y si ya existiera se borrará su contenido. Formato del Fichero:

<RANKING><ESPACIO><NOMBRE_EQUIPO><ENTER>

Ejemplo: fiba.txt

8 Italia

5 Serbia

2 España

4 Grecia

6 Lituania

7 Alemania

X. **Salir del Programa.** Se pedirá confirmación de salida.

Los ALUMNOS QUE NO TENGAN SUPERADO EL TRABAJO EN CLASE (tienen una nota acumulada < 2), además tendrán que implementar las siguientes opciones del menú principal:

D. **Borrar Equipo por Nombre.** Se pedirá el nombre del equipo y se mostrará toda la información del equipo por pantalla. Posteriormente se pedirá confirmación del borrado y en caso de ser afirmativa se borrará del árbol. Caso de no existir un equipo con dicho nombre se informará del error.

E. **Cargar Equipos desde Fichero.** Se pedirá al usuario el nombre de un fichero de texto que tendrá el formato del apartado B y se cargará toda la información en memoria. Se deberá informar de cualquier error relativo al manejo de ficheros o de memoria. En caso de que ya esté en memoria un equipo el mismo nombre el leído se ignorará el error de equipo repetido..

Una vez completadas las operaciones básicas pedidas, PARA SACAR MÁS NOTA habrá que implementar el resto de opciones del menú:

F. Listar por Ranking. Se mostrará por pantalla un listado de todos los equipos ordenados por su ranking FIBA.

G. Borrar Equipo por Ranking. Se pedirá el ranking del equipo y se mostrará toda la información del equipo por pantalla. Posteriormente se pedirá confirmación del borrado y en caso de ser afirmativa se borrará del árbol. Caso de no existir un equipo con dicho ranking se informará del error.

Módulos a Implementar:(LA DEFINICIÓN DE LOS MÓDULOS NO SE PODRÁ MODIFICAR)

- MError: Manejo de errores.
 - Define el tipo TError como un enumerado con los posibles errores generados.
 - **MuestraError (error)**: Muestra el error por pantalla.
- MEquipo: Manejo del registro TEquipo.
 - Define las constantes SP (espacio) y ENTER.
 - Define el tipo TEquipo como un registro con los campos ranking y nombre.
 - **LeeEquipo (equipo)**: Lee un equipo (ranking y nombre) desde teclado.
 - **EscribeEquipo (equipo)**: Escribe un equipo (ranking y nombre) por pantalla.
 - **LeeEquipoFichero (fichero, equipo)**: Lee un equipo (ranking y nombre) desde un fichero de texto cuyo descriptor se pasa como parámetro.
 - **EscribeEquipoFichero (fichero, equipo)**: Escribe un equipo (ranking y nombre) en un fichero de texto cuyo descriptor se pasa como parámetro.
- MABB: Manejo de un Árbol binario de búsqueda de equipos cuya clave de inserción/búsqueda es el nombre del equipo.
 - Define los tipos TABB y TNode para representar un árbol binario de búsqueda.
 - **CrearABB**: Crea una TABB vacío.
 - **DestruirABB (arbol)**: Destruye un TABB.
 - **InsertarEquipo (arbol, equipo, error)**: Inserta un equipo en el árbol. La clave de inserción es el nombre del equipo. No podrá haber equipos repetidos y deberá informarse de cualquier error producido.
 - **ListarABB Nombre (arbol)**: Mostrará por pantalla un listado de los equipos ordenados alfabéticamente por nombre.
 - **SalvaABBFichero (nombreFichero, arbol, error)**: Almacenará toda la información del árbol en preorden en el fichero cuyo nombre se facilita. Se informará de cualquier error producido.
 - **ListarABB_Ranking (arbol)**: Mostrará por pantalla un listado de los equipos ordenados por su ranking.
 - **MaxRanking (arbol)**: Devolverá el mayor ranking de los equipos almacenados en el árbol.
 - **BuscarEquipoRanking (arbol, ranking, equipo, error)**: Devolverá el equipo cuyo ranking se facilita. En caso de no existir un equipo con dicho ranking se devolverá el error correspondiente.
 - **BuscarEquipoNombre (arbol, nombre, equipo, error)**: Devolverá el equipo cuyo nombre se facilita. En caso de no existir un equipo con dicho nombre se devolverá el error correspondiente.
 - **EliminarEquipoNombre (arbol, nombre, error)**: Eliminará del árbol el equipo cuyo nombre se facilita. En caso de no existir un equipo con dicho nombre se devolverá el error correspondiente.

NOTAS

1. El directorio de trabajo será C:\LPGSEP09
2. OPCIONES A, B y C (mínimo para aprobar para quien tenga superada la nota de clase) 3 puntos. Nótese que están en negrita y subrayado los procedimientos y funciones necesarios para implementar esas opciones.
3. OPCIONES D, y E (sólo deben hacerla quien no tenga superada la nota de clase) 2 puntos
4. OPCIONES F, G y calidad de la implementación (opciones para subir nota) 5 puntos
5. **El uso de detalles de implementación del árbol fuera del módulo de implementación** del mismo (MABB.cpp) será **CAUSA de SUSPENSO**, es decir, en el programa principal no se podrá hacer cosas tipo: a= NULL, ptr=ptr->izq, etc.
6. **Añadir procedimientos o funciones a la definición de un módulo** será **CAUSA de SUSPENSO**
7. Todo **PROGRAMA QUE NO COMPILE** o tenga **efectos laterales** se considerará **SUSPENSO**.
8. No se distinguirá entre mayúsculas y minúsculas.
9. Se recomienda y valora el tratamiento de errores y la buena descomposición del programa principal en procedimientos y funciones, así como el uso de procedimientos y funciones auxiliares dentro de la implementación de los módulos cuando estas sean necesarias.
10. Recordad que todo fichero de cabecera debe llevar su pragma #ifndef #endif . Por ejemplo:

```
#ifndef _MError_h_
#define _MError_h_
. . .
#endif
```