



UNIVERSIDAD
DE MÁLAGA

Departamento de Lenguajes
y Ciencias de la Computación
UNIVERSIDAD DE MÁLAGA

LABORATORIO DE PROGRAMACIÓN

E.T.S.I. Informática- GESTIÓN/SISTEMAS. 20-JUNIO-2012

Duración del Examen: 3.5 horas

APELLIDOS _____ NOMBRE _____

ORDENADOR _____

La UEFA nos ha pedido la realización de un programa para el almacenamiento y asignación de árbitros para la próxima Eurocopa. El número de árbitros que se almacenarán está indeterminado, sin que podamos establecer un número máximo. De cada árbitro sabemos su número de colegiado (número natural), su nombre y apellidos (cadenas de caracteres) y su zona geográfica (un enumerado con los valores posibles norte, sur, este y oeste). El programa deberá presentar las siguientes opciones de menú:

```
Nombre: (Apellidos, Nombre)           Curso: 1°
Especialidad: (Gestión/Sistemas)
Puesto: número de ordenador           Fecha: 20/06/12

ARBITROS EUROCOPA
=====
A. Insertar Arbitro desde Teclado.
B. Insertar Arbitros desde Fichero.
C. Borrar Arbitro.
D. Salvar Arbitros a Fichero
E. Listar Arbitros.
F. Asignar Arbitro a Partido.

X. Salir del Programa

Introduzca Opción:
```

A. Insertar Árbitro desde Teclado. En esta opción se pedirá por teclado el número de colegiado. Si ya existe un árbitro almacenado que tenga ese número de colegiado, se mostrarán todos sus datos y no se insertará. En caso de que no exista, se pedirá su nombre, apellidos y zona geográfica y se insertará en la estructura de forma ordenada por el número de colegiado.

B. Insertar Árbitros desde Fichero. En esta opción se pedirá al usuario el nombre de un fichero de TEXTO en el cuál estarán los árbitros almacenados en el formato adjunto. Se leerán e insertarán todos los árbitros cuyo número de colegiado no esté ya almacenado en la estructura. Si el fichero no existe o está vacío, no se hará nada.

C. Borrar Arbitro. En esta opción se pedirá al usuario un número de colegiado. Si está almacenado, se mostrará su información y se pedirá confirmación de borrado, que si es afirmativa, borrará el árbitro y si es negativa no lo hará. En caso de solicitar el borrado de un árbitro que no esté almacenado, se informará de que el árbitro no está almacenado y no se hará nada.

D. Salvar Árbitros a Fichero. En esta opción se solicita al usuario el nombre de un fichero de texto del que se descartará su contenido, si existiera, y se grabarán todos los datos de todos los árbitros en el formato adjunto.

E. Listar Árbitros. En esta opción se muestran todos los árbitros almacenados, ordenados por su número de colegiado.

F. Asignar Arbitro a Partido. En esta opción se pedirá al usuario el nombre de los 2 equipos y la zona geográfica de cada uno de los equipos. El sistema asignará como árbitro al colegiado con más experiencia (el que tenga un número de colegiado más bajo) que no pertenezca a ninguna de las zonas geográficas de los equipos que juegan el partido. Se mostrarán por pantalla todos los datos del árbitro asignado y se eliminará dicho árbitro de la estructura.

X. Salir del Programa. En esta opción se pedirá confirmación de salida. Si es afirmativa, se terminará el programa, liberando todos los recursos que hayan sido reservados y si es negativa, se volverá al menú principal.

Formato del Fichero:

<número_colegiado> ESPACIO <nombre>#<apell1>#<apell2>#<zona>

Ejemplo: Arbitros.txt

98 Antonio Jesús#López#Nieto#sur
123 Arturo#Daudén#Ibáñez#norte

Módulos a Implementar, además del programa principal (eurocopa.cpp):

- MArbitro(MArbitro.h, MArbitro.cpp): Módulo de manejo de un árbitro.
 - Define las constantes ENTER('\n'), ESPACIO(' ') y SEPARADOR('#').
 - Define los tipos TNatural(unsigned int), TZona (Enumerado con valores: norte, sur, este, oeste) y TArbitro(registro con la información de un árbitro).
 - PintaArbitro: Escribe los datos de un árbitro por pantalla.
 - Zona_a_cadena: Devuelve la cadena que representa a una zona.
 - Cadena_a_zona: Devuelve la zona representada por una cadena.
 - SUC: nos retorna el sucesor de una zona.
 - LeeArbitro: Lee desde teclado la información de un árbitro del que ya sabemos su número de colegiado.
 - LeeZona: Lee una zona desde teclado.
 - LeeArbitroFichero: Lee toda la información de un árbitro desde un fichero de texto cuyo manejador se pasa como parámetro.
 - EscribeArbitroFichero: Escribe toda la información de un árbitro en un fichero de texto cuyo manejador se pasa como parámetro.
- MLista(MLista.h, MLista.cpp): Módulo de manejo de una LISTA ENLAZADA ORDENADA por número de colegiado.
 - Define el tipo TListaArbitros como una estructura dinámica.
 - CrearListaArbitros: Crea una lista nueva.
 - DestruirListaArbitros: Destruye la lista que se le pasa como parámetro.
 - InsertarArbitro: Mete un arbitro en la lista. Si ya estuviera almacenado un árbitro con el mismo número de colegiado, entonces ignora la inserción.
 - SacarMasAntiguoArbitro: Saca de la lista el árbitro con menor número de colegiado.
 - BorrarArbitro: Borra un árbitro de la lista, dado su número de colegiado.
 - ListaArbitrosLlena: Nos dice si una lista está llena.
 - ListaArbitrosVacía: Nos dice si una lista está vacía.
 - InsertaArbitrosFichero: Lee todos los árbitros que haya en un fichero de texto cuyo nombre se le pasa como parámetro y los inserta en la lista.
 - SalvaArbitrosFichero: Salva todos los árbitros que haya en la lista en un fichero de texto cuyo nombre se le pasa como parámetro. Si el fichero no existiera se creará y si ya existiera se borrará su contenido.

EL AÑADIR OPERACIONES QUE NO ESTÁN AQUÍ ESPECIFICADAS A LA DEFINICIÓN DE UN MÓDULO, ASÍ COMO EL USO DE DETALLES DE IMPLEMENTACIÓN DE LOS MÓDULOS DENTRO DEL PROGRAMA PRINCIPAL SERÁ CAUSA DE SUSPENSO.

Notas:

1. Es obligatorio trabajar en el directorio **Mis Documentos\LPJUN12**. Si no existe se creará.
2. Para **Aprobar** deberá ser correcta la definición de tipos, la modularización y funcionar **CORRECTAMENTE** las opciones A,B,C, D y X del menú.
3. Recordad que todo fichero de cabecera debe llevar su pragma #ifndef ... #endif

Ejemplo: Fichero: cabecera.h

```
#ifndef _Cabecera_h_
#define _Cabecera_h_
// Contenido del archivo de Cabecera
#endif
```