



ETSII LAB. PROGRAMACIÓN II
15 de Septiembre de 2000

DPTO. DE LENGUAJES Y CIENCIAS DE LA COMPUTACIÓN
ESCUELA TÉCNICA SUPERIOR DE INGENIEROS EN INFORMÁTICA
UNIVERSIDAD DE MÁLAGA

Apellidos y nombre:

Curso:

Grupo:

☞ *Lea detenidamente el examen antes de comenzar.*

☞ *No olvide escribir nombre, apellidos y grupo al que pertenece.*

Diseñar un programa en Modula-2 para gestionar datos referentes a los participantes en la Vuelta Ciclista al Campus Universitario por etapas, mediante el siguiente menú de opciones:

VUELTA CICLISTA AL CAMPUS

MENU

- A.- Volcar desde fichero.
- B.- Visualizar datos.
- C.- Actualización de etapa.
- D.- Baja.
- E.- Volcar a fichero.
- F.- Salir.

Elegir opción:

Opción A: se pedirá un nombre de fichero por teclado con la información de los participantes en la Vuelta Ciclista al Campus Universitario y se creará la estructura de datos que mantiene a la Vuelta Ciclista (si ésta estructura ya existía debe destruirse previamente). El formato del fichero (que será de texto) debe ser el siguiente:

<Nombre del participante>

<Equipo al que pertenece>

<Dorsal>

<Tiempo total acumulado> **expresado en el formato hh/mm/ss

El fichero de texto debe de crearlo el usuario con el editor de texto, y **por simplicidad siempre se supondrá que los datos de dicho fichero son correctos**. En caso de no existir dicho fichero en el momento de la lectura, se dará un mensaje de error y volverá a presentar el menú principal.

El número de corredores de cada equipo es **variable**, pero el número de equipos es **fijo** y son:

IGest, ISist, ISup, TelecoSup, TelecoTec

¡Ojo!: La estructura de datos que mantiene la Vuelta debe elegirse adecuadamente para que cumpla los requisitos.

Opción B: se pide visualizar la información de los corredores agrupados por equipos, manteniendo el orden anterior de equipos.

Opción C: con esta opción actualizaremos el tiempo empleado por cada ciclista en el total de etapas recorridas hasta el momento. Se pedirá el nombre del ciclista, y el tiempo empleado por éste en la etapa, acumulando dicho valor en el cómputo global del corredor. En caso de no existir el corredor en la estructura, se deberá de informar con un mensaje de error y volver al menú principal. Se deberá comprobar el formato correcto del tiempo a acumular.

_____ dar de baja un corredor. Se pedirá por teclado el dorsal, y los datos del corredor pasarán a ser eliminados de la estructura. Al igual que en la opción anterior, en caso de no existir el dorsal, el programa escribirá un mensaje de error y volverá al menú principal.

_____ se pedirá el nombre de un fichero y se volcará toda la información en él, exactamente con el mismo formato del fichero de lectura. Si el fichero ya existiese, se dará un mensaje por pantalla de aviso de fichero ya creado. Entonces el usuario debe decidir si la información del fichero debe ser actualizada o no. Una vez realizado el volcado, **la información de la estructura de datos en memoria deberá ser destruida en su totalidad.**

_____ salir del programa confirmando dicha salida (y liberando toda la memoria).

Para realizar el programa, se pide usar una librería que contenga las funciones y procedimientos lista. Al menos debe contener los siguientes procedimientos:

CrearLista (*crea la lista*),
DestruirLista (*Destruye una lista liberando la memoria *)
InsertarenLista (*Inserta un elemento. La lista no tiene que ser ordenada*)
BorrarenLista (* Borra un elemento de la lista *)
IterarLista (*Devuelve el elemento que se encuentre en la posición i de la lista*)
Longitud (*Devuelve la longitud de la lista *)

Recomendación: Implementar inicialmente el TAD lista (el cual será **opaco**; la estructura y el tratamiento de errores se deja al criterio del alumno pero se recomienda buscar la simplicidad). Posteriormente implementar el programa VUELTA.MOD.

Notar que la estructura que mantiene la vuelta **NO** tiene que ser necesariamente opaca.

☞Notas importantes:

- ☞NO se puede hacer uso de la librería Str.
- ☞ NO se distinguen mayúsculas y minúsculas. Pepe=pePE.
- ☞ LOS FICHEROS TENDRÁN LOS NOMBRES: LISTA.DEF, LISTA.MOD Y VUELTA.MOD, Y DEBEN DE ESTAR UBICADOS EN EL DIRECTORIO C:/TS/LP2BCG.
SE ENTREGARÁN TODOS LOS FICHEROS FUENTES Y EL EJECUTABLE.
- ☞ RECUERDE CODIFICAR LOS PROCEDIMIENTOS DE CREACIÓN Y DESTRUCCIÓN CUANDO SEAN NECESARIOS.
- ☞ LAS OPERACIONES DE CADENAS PUEDEN MODULARIZARSE EN MODULOS (EJ.CADENAS.DEF Y CADENAS.MOD) O BIEN INTEGRARSE EN EL FICHERO PRINCIPAL VUELTA.MOD
- ☞ SE VALORARÁ LA SIMPLICIDAD DEL CÓDIGO Y EL SANGRADO CORRECTO DEL PROGRAMA.
- ☞ COMPRUEBE LA CORRECTA GRABACIÓN DE LOS FICHEROS EN SU DISCO.