



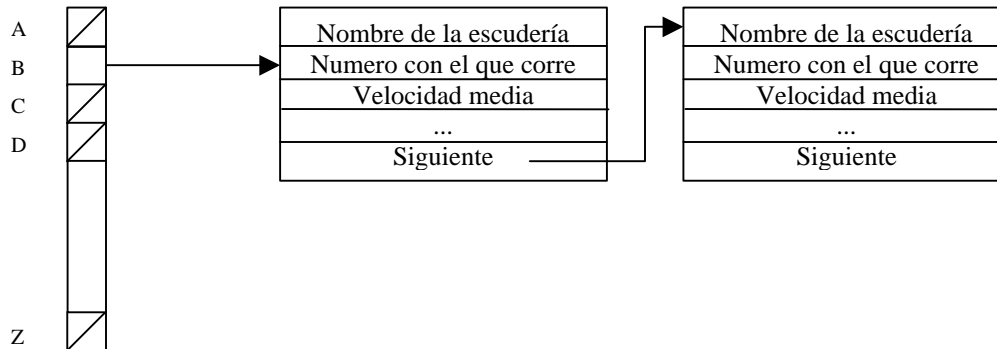
**E.T.S.I. Técnicos en Informática de Sistemas**  
**Laboratorio de Programación II. 1ºB**  
**Práctica 7: Estructuras de Datos Avanzadas (3)**  
**2000/2001**

Se requiere implementar un programa para simular un *rally* de coches de carreras en un circuito cerrado. Para ello, se desea elaborar un tipo abstracto de datos ópaco que será usado en un módulo de programa escritos en MODULA-2 que ejecutará un menú para que el usuario simule la carrera.

**Requisitos:**

El tipo abstracto de datos se llamará *Rally* (ficheros *RALLY.DEF* y *RALLY.MOD*) y presentará la estructura que se indica a continuación (la definición **debe ser opaca**).

La estructura está compuesta de un vector con índice desde la 'a' a la 'z' en el que se encuentran enlazados ordenados alfabéticamente por el nombre de la escudería los coches participantes. Cada registro contendrá el **nombre** de la escudería a la que pertenece el coche (15 caracteres como máximo), el **número** con el que participa en la carrera, la **velocidad media** a la que en este momento está circulando el coche y finalmente, **cualquier información** que el programador estime *razonablemente necesaria*.



Para este tipo de datos debe elaborarse un conjunto de operaciones completo que permita ejecutar el siguiente menú (pase todo como parámetros y no realice efectos secundarios en la definición del tipo, además evite dar mensajes por pantalla desde los módulos de implementación del tipo abstracto):

A.- Introducir nuevo coche.  
 B. Ganador en este momento.  
 C. Cambiar velocidad media.  
 D. Visualizar todo.  
 E. Acelerar algunas escuderías.  
 F. Salir.  
 >>>>>>>>>>Seleccione una opción.

**OPCION A**

Se debe leer desde teclado todos los datos respecto a un coche (escudería, número y velocidad media inicial en m/s). La inserción en la estructura se hace primero según por la letra en que empieza la escudería y dentro de cada letra **ordenados alfabéticamente** por el campo escudería completo. Considere iguales las mayúsculas y las minúsculas.

No olvide que es imprescindible comprobar la **validez de los datos leídos por teclado** (velocidad positiva, número positivo, sólo un coche por escudería, números no repetidos).

**OPCION B**

Esta opción permite informar al usuario cual es el coche que va en primera posición. Para ello pedirá el tiempo transcurrido desde la última vez que se usó esta opción. Dado que el circuito tiene 1000 metros se calculará para cada coche el número de vueltas dadas (puede ser un valor no entero) y el que más vueltas haya dado es el ganador en este momento. Si uno o varios coches han dado más de 100

vueltas entonces el ganador del *rally* es aquel cuyo número de vueltas es el mayor de entre los obtenidos (pueden darse empates y por tanto existir varios ganadores).

#### **OPCION C**

Esta opción permite cambiar en cualquier momento la velocidad media de cualquiera de los coches. El menú debe pedir la escudería, el número y la nueva velocidad media del coche en la misma línea. Si cualquiera de estos datos (escudería o número) no aparecen juntos para alguno de los coches almacenados entonces se debe dar un mensaje de error en pantalla.

#### **OPCION D**

Esta opción debe presentar de forma ordenada alfabéticamente todos los coches que participan en la carrera (no muestre datos vacíos, sólo los que haya almacenado el usuario). Haga pausas en la salida para poder leerla cómodamente.

#### **OPCION E**

Esta opción pedirá al usuario un nombre de fichero de texto donde después se debe almacenar el contenido de toda la estructura de datos de una manera legible. El formato del fichero de texto se deja a la libre elección del alumno pero recuerde que debe ser adecuado para su posterior lectura en la opción F.

#### **OPCION F**

Esta opción permite cargar desde un fichero de texto (cuyo nombre se pide por teclado) un nuevo contenido para la estructura de datos que reemplazará al contenido actual (debe destruirse antes todo el contenido de la estructura). Si el fichero no existe debe darse un mensaje de error.

#### **OPCION G**

Esta opción permite al usuario teclear un valor positivo que se interpretará como un incremento en la velocidad de todos los coches cuya escudería comience por una letra entre la 'a' y la 'm' y cuyo número de participante tenga tres o más dígitos.

#### **OPCION H**

Esta opción se usará para terminar la ejecución del programa. Recuerde preguntar antes si desea abandonar realmente (S/N) y en caso afirmativo el programa debe escribir en pantalla el nombre completo del alumno, su grupo y la máquina en que se elaboró el examen.

El alumno habrá de codificar en Modula-2 el módulo de definición y el de implementación del TAD anteriormente definido, llamado **RALLY.MOD** y **RALLY.DEF**.

Aprobado: A, B, C, D y H (Sin prácticas además la opción G).

Notable: Si funcionan todas las opciones del menú.

Sobre. Se valora la claridad, estilo y comentarios.

#### **OBSERVACIONES**

1. Recuerde que en la parte de definición de un TAD únicamente han de estar presentes las operaciones propias del mismo. Si se necesitan procedimientos auxiliares habrán de aparecer únicamente en el módulo de implementación. No defina más de un tipo abstracto por módulo.
2. Nunca utilice submenús para pedir datos. Use exclusivamente el menú principal dado y después lea los datos según la opción elegida. No incluya ninguna pregunta del tipo "¿Desea incluir más datos?".
3. No se puede usar las funciones de la librería STR y recuerde que para aprobar hay que modularizar (uso profuso de procedimientos y funciones), evitar usar variables globales.
4. Compruebe la correcta grabación de los ficheros en el disco.