

Ejercicios sobre estructuras. Clase 06/05/2014

1. Se va a utilizar un programa para tratar los datos de los N clientes de un gimnasio. De cada cliente se van a considerar los siguientes datos: **nombre**, **NIF**, **edad**, **estatura** (en metros) y **peso**.
 - Diseña las estructuras de datos necesarias para mantener en memoria esta información de todos los clientes.
 - Escribe un programa que realice las siguientes operaciones:
 - ▶ `leeClientes`: Leer del teclado los datos de los clientes y guardarlos en memoria.
 - ▶ `clienteMasJoven`: Obtener los datos del cliente más joven.
 - ▶ `imcMaxMin`: Obtener los datos de los clientes con menor y con mayor índice de masa corporal. El *IMC* se calcula de la siguiente forma:

$$IMC = \frac{peso}{estatura^2}$$

Utiliza subprogramas para realizar las distintas funciones. Por ejemplo, para mostrar en pantalla los datos de un cliente puedes utilizar un subprograma con el siguiente prototipo:

```
void mostrarCliente(tCliente cl);
```

Ejercicios sobre estructuras. Clase 06/05/2014

2. Se va a extender el programa anterior para mantener información sobre las puntuaciones obtenidas en las pruebas que organiza el gimnasio (*spin*, *pilates*, *body pump*, etc). De cada cliente se deben mantener las **puntuaciones** de las K **pruebas** realizadas (las pruebas están numeradas de 0 a $K - 1$). Todos los clientes tienen puntuación en todas las pruebas.
 - Modifica las estructuras de datos del ejercicio anterior para mantener en memoria toda la información.
 - Escribe un programa que realice las siguientes operaciones sobre los datos de clientes:
 - ▶ Modificar la lectura de datos de los clientes para incluir puntuaciones.
 - ▶ `clienteMax`: Obtener el cliente que tiene la puntuación más alta de una prueba determinada.
 - ▶ `medMaxMin`: Obtener la puntuación media, máxima y mínima de cada una de las pruebas.
 - ▶ `mediaMax25`: Obtener el cliente que ha obtenido la puntuación media más alta de los clientes con *IMC* mayor a 25.