Vamos a hacer un ejemplo completo que use tablas (“arrays”), registros (“struct”) y que además manipule cadenas.

La idea va a ser la siguiente: Crearemos un programa que pueda almacenar datos de hasta 1000 ficheros (archivos de ordenador). Para cada fichero, debe guardar los siguientes datos: Nombre del fichero (max 40 letras), Tamaño (en KB, número de 0 a 2.000.000.000). El programa mostrará un menú que permita al usuario las siguientes operaciones:

1- Añadir datos de un nuevo fichero  
2- Mostrar los nombres de todos los ficheros almacenados  
3- Mostrar ficheros que sean de más de un cierto tamaño (por ejemplo, 2000 KB).  
4- Ver todos los datos de un cierto fichero (a partir de su nombre)  
5- Salir de la aplicación (como todavía no sabemos almacenar los datos, éstos se perderán).

No debería resultar difícil. Vamos a ver directamente una de las formas en que se podría plantear y luego comentaremos alguna de las mejoras que se podría (incluso se debería) hacer.

Una opción que podemos a tomar para resolver este problema es la de contar el número de fichas que tenemos almacenadas, y así podremos añadir de una en una. Si tenemos 0 fichas, deberemos almacenar la siguiente (la primera) en la posición 0; si tenemos dos fichas, serán la 0 y la 1, luego añadiremos en la posición 2; en general, si tenemos “n” fichas, añadiremos cada nueva ficha en la posición “n”. Por otra parte, para revisar todas las fichas, recorreremos desde la posición 0 hasta la n-1, haciendo algo como

for (i=0; i<=n-1; i++) { ... más órdenes ...}

o bien algo como

for (i=0; i<n; i++) { ... más órdenes ...}

El resto del programa no es difícil: sabemos leer y comparar textos y números. Sólo haremos dos consideraciones:

* No se comportará correctamente si los textos (nombre del fichero, por ejemplo) contienen espacios, porque aún no sabemos leer textos con espacios.
* Hemos limitado el número de fichas a 1000, así que, si nos piden añadir, deberíamos asegurarnos antes de que todavía tenemos hueco disponible.