

## Ejercicio – 2 de abril de 2013

Se pide construir un programa (**main.cpp**) que procese el stock de una tienda.

El stock es una lista de longitud variable de productos (máximo 100). Para cada producto se mantiene la siguiente información:

- ✓ Código (entero positivo)
- ✓ Nombre
- ✓ Precio
- ✓ Unidades existentes
- ✓ Lista de entradas y salidas
- ✓ Actualizado (stock ya actualizado o no)

Cada producto tiene su propia lista de entradas y salidas, que también es una lista de longitud variable en la que cada elemento tiene la siguiente información:

- ✓ Fecha (cadena con formato AA/MM/DD)
- ✓ Tipo (carácter: E para entrada y S para salida)
- ✓ Unidades que entran o salen

Toda la información se guarda en un archivo, denominado **datos.txt**, con la siguiente estructura:

```
Código de producto (-1 termina)
Nombre del producto
Precio del producto
Unidades existentes
Fecha (X termina)
Tipo (E o S)
Unidades
Fecha (X termina)
Tipo (E o S)
Unidades
...
X
Código de producto (-1 termina)
...
-1
```



El programa deberá incluir los siguientes subprogramas:

- ✓ `leer()`: carga la información del archivo `datos.txt` en la lista de productos; devuelve `true` si se ha podido abrir el archivo y `false` en caso contrario.
- ✓ `ordenar()`: ordena las listas de entradas y salidas por orden de fecha (menor a mayor).
- ✓ `buscarProducto()`: dada la lista de productos y un código, devuelve la posición (índice) del elemento de la lista de productos con ese código; si no se encuentra devuelve -1.
- ✓ `buscarES()`: dada una lista de entradas y salidas, y una fecha, devuelve la posición (índice) del elemento de la lista con esa fecha; si no se encuentra devuelve -1. Debe implementarse como búsqueda binaria, usándose sólo tras ordenar las sublistas.
- ✓ `mostrar()`: dada la lista de productos, muestra la información del stock.
- ✓ `actualizar()`: dada la lista de productos, actualiza todas las unidades existentes de productos, recorriendo cada lista de entradas y salidas, y aplicando la entrada o salida de unidades; si en algún momento las unidades se hacen negativas se notificará el error.

El programa probará los distintos subprogramas en la función `main()`.

Entrega, a través de la tarea **04-02** del Campus Virtual, sólo el archivo con el código fuente (**`main.cpp`**). ¡No olvides poner al principio del programa un comentario con los nombres y el número del grupo!

