

Convocatoria ordinaria**Tiempo disponible: 1 hora – Puntuación: 6,5 puntos**

Se quiere desarrollar una aplicación para gestionar las ventas de los productos ofertados en una pescadería.

Para representar la información de cada producto se define un tipo `tProducto` que se compone de:

- ✓ La denominación del producto (cadena de caracteres sin espacios en blanco), que lo describe de manera unívoca.
- ✓ La cantidad o peso (Kg.) de producto disponible para ser vendido.
- ✓ El precio de venta por Kg. de producto.

La aplicación debe disponer de un catálogo de productos. Para ello se define un tipo `tListaProductos` que representa una lista de hasta 50 punteros a variables de tipo `tProducto`. El catálogo estará ordenado de manera decreciente por la denominación de los productos. A modo de ejemplo, aquí tenemos un catálogo de 7 productos:

Salmonete	Merluza	Mejillón	Lubina	Gambón	Gamba	Besugo
10,50 kg	30,00 kg	80,00 kg	80,00 kg	10,00 kg	50,50 kg	20,50 kg
15,50€/kg	12,00€/kg	4,00€/kg	13,50€/kg	40,0€/kg	25,95€/kg	20,95€/kg

A través de la aplicación un cliente debe poder realizar su compra. Definimos un tipo `tCesta` que representa la cesta de la compra de un cliente: una lista implementada con un array dinámico con 20 elementos como máximo, donde cada elemento es un valor del tipo `tPetición` compuesto de la denominación del producto y la cantidad (en Kg.) que se desea comprar. La cesta de la compra carece de orden. A modo de ejemplo, aquí tenemos el contenido de una lista de la compra con 4 peticiones:

Mejillón	Dorada	Merluza	Gambón
10,00 kg	1,00 kg	3,00 kg	20,00 kg

En estos momentos tenemos definidos los tipos y hemos implementado un subprograma para cargar desde archivo el contenido del catálogo de productos, otro para cargar el de una cesta de la compra, y sendos subprogramas para mostrar un catálogo y una cesta por pantalla. También tenemos un pequeño programa principal de prueba que deberá acabar haciendo lo siguiente: (a) cargar un catálogo y una

cesta de la compra, (b) visualizarlos por pantalla, (c) invocar a un subprograma `hazLaCompra` que permite realizar la compra y calcular el coste de la misma, (d) visualizar por pantalla tanto el coste de la compra como el estado en el que queda el catálogo tras la compra, y (e) liberar la memoria dinámica empleada. En estos momentos el programa principal hace los pasos (a) y (b).

Tu misión es completar el primer prototipo de esta aplicación. Para ello debes:

1. Desarrollar el subprograma `hazLaCompra`, que recibe un catálogo de productos y una cesta de la compra y realiza las ventas que son posibles, computando y devolviendo el coste total de la compra y dejando actualizado el catálogo de productos. A la hora de realizar las ventas ten en cuenta que:
 - La cesta de la compra puede incluir productos que no existen en el catálogo, no pudiendo adquirirse.
 - En la cesta de la compra puede solicitarse una cantidad de un cierto producto superior a la disponible en el catálogo, pudiéndose realizar la venta sólo de la cantidad disponible.
 - La realización de la venta de cada petición debe actualizar convenientemente la cantidad de cada producto solicitado que queda disponible en el catálogo, teniendo en cuenta que si se agota un producto desaparecerá de la lista (sin destruir el orden que hay en ella).

Tomando como ejemplo el catálogo y la cesta anteriores, el coste total a mostrar sería 476€ (resultado de $10,00 \text{ kg} \times 4,00 \text{ €/kg} + 3,00 \text{ kg} \times 12,0 \text{ €/kg} + 10,00 \text{ mkg} \times 40,00 \text{ €/kg}$ ¡Ojo!: fíjate que sólo se pueden comprar 10 Kg. de Gambón y no los 20 solicitados, y que no hay el producto Dorada en el catálogo) y el catálogo quedaría actualizado así (puedes ver que ha desaparecido el producto Gambón):

Salmonete	Merluza	Mejillón	Lubina	Gamba	Besugo
10,50 kg	27,00 kg	70,00 kg	80,00 kg	50,50 kg	20,50 kg
15,50€	12,00€	4,00€	13,50€	25,95€	20,95€

2. Desarrollar un subprograma para liberar la memoria dinámica empleada por un catálogo de productos y otro para liberar la empleada por una cesta de la compra.
3. Completar el código del programa principal para que realice los pasos (c), (d) y (e).

Para el catálogo y la cesta de este enunciado (que son los que hay en los archivos .txt dados como documentación) la ejecución deberá ser la siguiente:

```
Consola de depuración de Microsoft Visual Studio

***** CATALOGO DE PRODUCTOS *****
Salmonete - 10.5 kg - 15.5 euros
Merluza - 30 kg - 12 euros
Mejillon - 80 kg - 4 euros
Lubina - 80 kg - 13.5 euros
Gambon - 10 kg - 40 euros
Gamba - 50.5 kg - 25.95 euros
Besugo - 20.5 kg - 20.95 euros

***** CESTA DE LA COMPRA *****
Mejillon - 10 kg
Dorada - 1 kg
Merluza - 3 kg
Gambon - 20 kg

Y ahora veamos cuanto cuesta la compra y como queda el catalogo

El coste de la compra asciende a 476 euros
***** CATALOGO DE PRODUCTOS *****
Salmonete - 10.5 kg - 15.5 euros
Merluza - 27 kg - 12 euros
Mejillon - 70 kg - 4 euros
Lubina - 80 kg - 13.5 euros
Gamba - 50.5 kg - 25.95 euros
Besugo - 20.5 kg - 20.95 euros
```

