

Fundamentos de la programación – Grupos D, G, H

Curso 2014–2015

Control de febrero

Tiempo disponible: 3 horas

Se necesita desarrollar una aplicación que permita gestionar los pedidos online de un supermercado. La aplicación tiene almacenada la siguiente información acerca de cada producto en venta: código del producto, descripción, precio por unidad y si está o no en oferta.

La aplicación debe manejar un catálogo de NUM_PRODUCTOS (ni uno más ni uno menos) donde cada elemento representa la información de un producto. Ejemplo de catálogo con NUM_PRODUCTOS = 5:

L0001	H0025	PD010	Y8912	AOVE8
Leche	Huevos	Pasta de Dientes	Yogures	Aceite Oliva
1.25	1.50	3.15	1.50	4.60
false	false	false	true	false

A través de la aplicación, un cliente podrá realizar un pedido incluyendo productos en la cesta. La cesta será una lista de productos que tendrá entre 0 y MAX_CESTA elementos. Cada elemento de esta lista guarda el código del producto y las unidades solicitadas por el cliente. Ejemplo de cesta:

Y8912	AOVE8	L0001
2	1	6

Realiza la declaración de tipos apropiada para representar la información descrita.

Programa principal

El programa comenzará cargando la información de `productos.txt` en el catálogo. Si la carga tiene éxito se mostrará un menú con estas opciones:

1. Añadir productos a la cesta.
2. Confirmar pedido.
0. Salir

Se permitirá realizar cada una de ellas tantas veces como sea necesario hasta que el cliente decida salir.

Cargar datos

Los datos del catálogo de productos se cargarán del archivo `productos.txt`. Si hay más de `NUM_PRODUCTOS` en el archivo, los excedentes se ignoran. El archivo `productos.txt` está organizado de manera que la información de cada producto aparece en una línea: primero el código del producto, luego el precio por unidad, a continuación si está o no en oferta (1 para cierto y 0 para falso) y, por último, la descripción del producto. El archivo finaliza con `XXX` como centinela. Ejemplo de `productos.txt`:

```
L0001 1.25 0   Leche
PC010 3.15 0   Pasta de dientes
H0025 1.50 0   Huevos
Y8912 1.50 1   Yogures
XXX
```

Debes implementar un subprograma **cargar** que genere el catálogo con los productos en venta cargado del archivo `productos.txt` y devuelva un booleano que indique si la carga se ha podido realizar con éxito o no.

Añadir productos al pedido

Si el cliente elige la opción 1 del menú, se mostrarán los productos del catálogo y se leerán por teclado los códigos de los productos que el cliente desea incluir en la cesta, así como las unidades de cada uno de ellos. Se deberán realizar las siguientes comprobaciones:

1. Si el cliente solicita un producto que no se encuentra en el catálogo, se le notificará que no se dispone de dicho producto y no se incluirá en la cesta.
2. Si el producto seleccionado ya se encuentra en la cesta, se informará al usuario indicando el número de unidades de dicho producto ya incluidas en la cesta, y se preguntará si desea actualizar el número el número de unidades o anular la petición.

Para esta opción se incluirán, al menos, los siguientes subprogramas:

- ✓ **mostrarProductos**: Dado el catálogo de productos en venta, muestra por pantalla la información (código, descripción, precios, si está o no en oferta) de todos los productos.
- ✓ **llenarCesta**: Produce una cesta (lista de productos seleccionados) leída de teclado. Se solicita al cliente el código y las unidades de los productos que desea comprar. Cuando el cliente no quiera incluir más productos en la cesta, introducirá `XXX` como código del producto.
- ✓ **estaEnCatalogo**: Dado el catálogo de productos en venta y el código de un producto, devuelve si se encuentra o no en el catálogo.
- ✓ **posCesta**: Dada una cesta y el código de un producto, devuelve la posición donde se encuentra el producto (o la primera posición libre en la cesta).

Confirmar pedido

Si el cliente elige la opción 2 del menú, se guardarán los productos de la cesta en el archivo `compra.txt`: código, unidades, e importe correspondiente (teniendo en cuenta si están o no en oferta). Las ofertas consisten en “dos por uno”. Por ejemplo, si se compran 6 unidades, se pagan 3, y si se compran 9 unidades, se pagan 5 (pues la novena va suelta). Ejemplo de `compra.txt`:

```
L0001      10    12.50
H0025       1     1.50
Y8912       4     3.00
XXX
```

Además, después se mostrará en pantalla el total de la compra.

Para esta opción se incluirá, al menos, un subprograma **precioLinea**, que dado un ítem de la cesta, devuelve su precio total teniendo en cuenta las unidades y si está o no en oferta.

Puedes ver un ejemplo de ejecución de todo el programa más abajo.

Instrucciones de entrega

- Pon al inicio de tu archivo `.cpp` un comentario con tu nombre completo y DNI.
- Sube el archivo al Campus Virtual mediante la opción destinada a ello.
- El archivo subido **debe compilar** y debe tener un **main** como el descrito en el enunciado.

Si no se cumplen estos requisitos no se corregirá el examen.

```
1.- Añadir productos a la cesta
2.- Confirmar pedido
0.- Salir
Opcion: 1

CODIGO      DESCRIPCION      PRECIO
L0001      Leche                1.25
PC010      Pasta de dientes       3.15
H0025      Huevos                 1.50
Y8912      Yogures                1.50 EN OFERTA
P0012      Pan de Molde           1.00
M2562      Mantequilla            2.45
D4567      Detergente              8.95
A0091      Aceitunas              1.20 EN OFERTA
CP987      Cafe                   2.75
M8907      Macarrones             1.20 EN OFERTA

Codigo Producto: L0001
Unidades: 10

Codigo Producto: M0968
El codigo indicado es erroneo

Codigo Producto: M8907
Unidades: 1

Codigo Producto: M8907
Unidades: 1

En la cesta hay 1 del producto seleccionado
¿ Desea incluir las unidades indicadas? (S/N): S

Codigo Producto: D4567
Unidades: 1

Codigo Producto: XXX

1.- Añadir productos a la cesta
2.- Confirmar pedido
0.- Salir
Opcion: 2

Importe Total: 22.65

1.- Añadir productos a la cesta
2.- Confirmar pedido
0.- Salir
Opcion: 0
Presione una tecla para continuar . . .
```