



Tecnologías de la Información



Universidad de Huelva

GRADO EN INGENIERÍA INFORMÁTICA

ESTRUCTURAS DE DATOS I

Práctica 1

Ficheros y Tablas Dinámicas

Versión 1.0.

Una empresa dispone de su información de productos y ventas asociadas almacenadas en ficheros cuya estructura se muestra a continuación:

Fichero de Productos

Fichero con sucesión de elementos de tipo

```
struct producto
{
    Sfecha ultimaventa;
    int unidades;
    int producto;
    float importe;
    int tipo;
    char nombre[20];
}
```

Siendo Sfecha una estructura como la siguiente:

```
struct Sfecha
{
    int dia;
    int mes;
    int anno;
};
```

En dicho fichero el producto de código i, ocupa la posición iésima del fichero. Si un producto ha sido eliminado o está en desuso, en el fichero igualmente está escrito su código, pero el resto de campo está sin uso, excepto el campo tipo que tiene un valor de -1 para denotar esa eliminación o desuso.

El valor del campo tipo corresponde a la siguiente codificación 1- Electrónica, 2 – Papelería o 3- Otros.

Fichero de Ventas

Fichero con sucesión de elementos de tipo

```
struct venta
{
    Sfecha fecha;
    int unidades;
    int producto;
    float importe;
};
```

En el campo fecha dispone de la fecha de la venta, las unidades vendidas, el importe de las mismas y el código del producto vendido. Las ventas están **ordenadas** ascendentemente por fecha (al principio del fichero las más antiguas, al final las más actuales).

Para el manejo de los ficheros descritos la empresa desea realizar un programa que se apoye en dos clases y que permita desarrollar las funcionalidades que a continuación se expresarán:

Clase ventas

```
class ventas
{
    fstream detalle;
    cadena fichero;
    cadena ficheroresumen;

    public:
        bool asignar(cadena Fichero,cadena FicheroResumen);
        void mostrarventas();
        void anadirventa(int num);
        void resumirfichero();
        void estadisticas(int tipo,int annoini,int annofin);
};
```

La clase ventas consta del atributo *fichero*, que es una cadena de caracteres que contiene el nombre del fichero asociado a las ventas, *ficheroresumen* es una cadena de caracteres que contiene el nombre del fichero físico asociado al maestro de productos, mientras que *detalle* es el nombre lógico del fichero asociado a las ventas.

Los métodos de la clase son:

bool asignar(cadena Fichero,cadena FicheroResumen);

método que recibe el nombre físico del fichero de ventas, el cual crea en caso de que no exista. Asigna de forma adecuada valor a los atributos detalle y fichero.

void mostrarventas();

Método que permite listar por pantalla la información de ventas recogida en el fichero de ventas en el formato:

```
Fecha de venta: DD/MM/YYYY
Producto:XXXX (Nombre del producto)
Unidades: YYYYYY
Importe: ZZZZZZ
-----
Fecha de venta: DD/MM/YYYY
Producto:XXXX (Nombre del producto)
Unidades: YYYYYY
Importe: ZZZZZZ
-----
...
```

void anadirventa(int num);

Método que recibe como parámetro el código de un producto y procede, tras completar la información de unidades, importe y fecha a introducir la venta en el fichero de ventas.

Importante resaltar que la venta debe introducirse de forma adecuada en el fichero, para que el mismo siga ordenado por fecha. Es posible que se introduzca una venta “retrasada” que no se hubiera introducido a su debido tiempo.

Con ello debe tener presente que la inserción de la venta puede producir desplazamientos en el fichero de ventas

void resumirfichero();

Método que permite actualizar el fichero de productos con la información de resumen de las ventas recogidas en el fichero de ventas. Con ello nos referimos que a partir del fichero de ventas se obtienen los totales de unidades e importes vendidos de cada producto, así como la última fecha de venta y se actualiza de forma adecuada el fichero de productos.

void estadisticas(int tipo,int annoini,int annofin);

Método que permite mostrar por pantalla listado de los productos más vendidos (atendiendo al número de unidades) del tipo de producto pasado por parámetro y entre los años comprendidos entre annoini y annofin.

Para ello, debe “resumirse” la información de las ventas que sean del tipo de producto indicado y comprendida en el intervalo pasado como parámetro y proceder a aplicar ordenación (para ello se le facilita método burbuja genérico, que deberá adaptar). Para dicho resumen a mostrar por pantalla, se desea usar expresamente una tabla dinámica donde depositar el resumen y posteriormente ordenar para mostrar por pantalla.

Clase productos

```
class productos
{
    int numproductos;
    int maxcatalogo;
    fstream resumen;
    cadena fichero;
    ventas ven;
    public:
        productos(cadena Fichero,cadena FicheroVentas);
        int getmaxcatalogo();
        void setmaxcatalogo(int num);
        int getnumproductos();
        void setnumproductos(int num);
        void mostrarproductos();
        void mostrarventas();
        bool anadirventa();
        void actualizarresumen();
        void obtenerestadisticas(int tipo,int annoini,int annofin);
};
```

Dicha clase tiene como atributos el número de productos “activos” (numproductos), el código mayor existente en el catálogo (maxcatalogo), resumen (nombre lógico para manejar el fichero de productos), fichero (nombre físico del fichero de productos) y ven, un objeto de clase ventas que contiene las ventas de la empresa.

Los métodos de la clase son:

productos(cadena Fichero,cadena FicheroVentas);

Constructor de la clase, que procede a abrir el fichero de productos (asociado al nombre físico pasado en el parámetro Fichero), rellenando de forma adecuada los parámetros maxcatalogo y numproductos. En caso de que no existiera fichero de productos previo, procede a crearlo. Al mismo tiempo dicho método asocia también el fichero de ventas de la empresa (cuyo nombre físico se pasa bajo el parámetro FicheroVentas).

int getmaxcatalogo();

Método que devuelve el maxcatalogo.

void setmaxcatalogo(int num);

Método que asigna el maxcatalogo.

int getnumproductos();

Método que devuelve el número de productos.

void setnumproductos(int num);

Método que asigna el número de productos.

void mostrarproductos();

Método que muestra por pantalla el fichero de productos en la forma:

Producto: XXXX (Nombre Producto)

Última Venta: DD/MM/YYYY

Importe: XXXXX

Unidades: ZZZZZZ

Tipo Producto: GGGGGGG

Producto: XXXX (Nombre Producto)

Última Venta: DD/MM/YYYY

Importe: XXXXX

Unidades: ZZZZZZ

Tipo Producto: GGGGGGG

....

void mostrarventas();

Método que sirve para mostrar listado de ventas de la empresa.

bool anadirventa();

Método que permite introducir una venta. Lo primero que hace el método es solicitar el código del producto. Si éste no existiera lo crea de forma adecuada en el fichero de productos. Posteriormente introduce la venta en el fichero de ventas.

void actualizarresumen();

Método que actualiza el fichero de productos con los totales de ventas (unidades e importe) así como con la fecha de la última venta de cada producto.

void obtenerestadisticas(int tipo,int annoini,int annofin);

Método que permite obtener un listado ordenado de los productos más vendidos (por número de unidades) del tipo de producto pasado por parámetro, entre los años *annoini* y *annofin*).

Programa principal

El programa a desarrollar deberá comenzar creando un objeto *productos* a partir de los ficheros físicos que se le proporciona, “*resumen.dat*” y “*ventas.dat*”, ficheros respectivamente de productos y de ventas.

Luego mostrará el siguiente menú:

```
Practica 1

1 ver fichero detalle ventas
2 ver fichero productos
3 Añadir venta
4 crear resumen ventas
5 Obtener estadísticas (tipo producto y periodo)
0 Salir
```

Las opciones 1 a 5 del menú se corresponden con llamadas a los métodos de la clase *productos*.

Ficheros proporcionados

Se proporcionan los ficheros *resumen.dat*, *ventas.dat* y *base.cpp*.

El fichero *base.cpp* se proporciona simplemente para recoger el código de ordenación *Burbuja* (que debe adaptar) y de otras constantes y tipos que se reflejan en el enunciado. Deberá hacer uso de dicho código y, en su caso, moverlo al módulo que estime más conveniente.

Y el fichero de productos y el fichero de ventas que se le proporcionan tienen el siguiente contenido:

Dpto. Tecnologías de la Información