

1)

1. Archivos Secuenciales

Una cadena de restaurantes posee un archivo de productos que tiene a la venta, de cada producto se registra: código de producto, nombre, descripción, código de barras, categoría de producto, stock actual y stock mínimo. Diariamente el depósito debe efectuar envíos a cada uno de los tres restaurantes que se encuentran en la ciudad de Laprida. Para esto, cada restaurante envía un archivo por mail con los pedidos de productos. Cada pedido contiene: código de producto, cantidad pedida y una breve descripción del producto. Se pide realizar el proceso de actualización del archivo maestro con los tres archivos de detalle, obteniendo un informe de aquellos productos que quedaron por debajo del stock mínimo y sobre estos productos informar la categoría a la que pertenecen. Además, informar aquellos pedidos que no pudieron satisfacerse totalmente por falta de stock, indicando la diferencia que no pudo ser enviada a cada restaurante. Si el stock no es suficiente para satisfacer un pedido en su totalidad, entonces el mismo debe satisfacerse con la cantidad que se disponga.

Nota: Todos los archivos están ordenados por código de producto

2)

1. Archivos Secuenciales

Dada la siguiente estructura:

Type

```
tProducto = record
    código: integer;
    nombre: string[50];
    presentacion: string[100];
end;
tArchProductos = file of tProducto ;
```

Las bajas se realizan apilando registros borrados y las altas reutilizando registros borrados. El registro 0 se usa como cabecera de la pila de registros borrados: 0 en el registro 0 implica que no hay registros borrados y N en el registro 0 indica que el próximo registro a reutilizar es el N, siendo éste un número relativo de registro válido. El encadenamiento de registros debe realizarse con el campo código de producto pero especificando los números de registro referenciados con signo negativo.

Implemente los siguientes módulos:

```
(Abre el archivo y agrega el producto recibido como parámetro manteniendo la política
descripta anteriormente)
procedure agregar (var a: tArchProductos; producto: tProducto);
```

```
(Abre el archivo y elimina el producto recibido como parámetro manteniendo la
política descripta anteriormente)
procedure eliminar (var a: tArchProductos; producto: tProducto);
```

3)

1. Archivos Secuenciales

Un supermercado tiene 25 cajas que registran diariamente las ventas de productos. De cada venta se dispone: número de ticket, código del producto y cantidad de unidades vendidas del producto.

Al finalizar el día, los archivos correspondientes a las cajas se ordenan por código de producto para luego actualizar el archivo de productos. Los registros del archivo de productos contienen el código del producto, la descripción, la cantidad en existencia, el stock mínimo y el precio de venta actual. Implementar un programa que permita:

- a) Dada la cantidad de cajas, actualizar la existencia de cada producto registrando la cantidad vendida en la jornada. Tenga en cuenta que el stock no puede quedar por debajo de cero.
- b) Informar aquellos productos que dispongan unidades en existencia y no hayan sido vendidos.
- c) Informar aquellos productos vendidos que quedaron por debajo del stock mínimo.
- d) Informar para cada código de producto, el nombre y el monto total vendido, y también informar el monto total facturado en el día para todos los productos.

NOTA: No debe implementar el ordenamiento de los archivos. Todos los archivos deben recorrerse una única vez

4)

1. El Gerente General de una cadena de librerías con sucursales en todo el país requiere información sobre ejemplares de libros vendidos en cada sucursal, totalizados por ISBN (*International Standard Book Number*) y por autor (puede haber más de un ISBN de un mismo autor).

Se dispone de un archivo con registros de ventas compuestos por código de sucursal (codSuc: word), identificador de autor (idAutor: longword), ISBN del libro (isbn: longword) e identificador interno de ejemplar (idEj: word). El archivo está ordenado por codSuc, idAutor e ISBN (puede haber varias ventas del mismo libro en una misma sucursal, pero de distintos ejemplares).

Definir los tipos de registro (tvta) y del archivo (tArchVtas) y codificar un procedimiento (totalizar) que reciba el archivo asignado y sin abrir y el nombre de un archivo de texto y reporte en el archivo de texto:

Código de Sucursal: ____

Identificación de Autor: ____

ISBN: ____ . Total de ejemplares vendidos del libro: ____

... (puede haber más ISBN del mismo autor)

Total de ejemplares vendidos del autor: ____

... (puede haber más autores)

Total de ejemplares vendidos en la sucursal: ____

... (puede haber más sucursales)

TOTAL GENERAL DE EJEMPLARES VENDIDOS EN LA CADENA: ____

5)

1. Archivos Secuenciales

Una cadena de tiendas de indumentaria posee un archivo maestro **no ordenado** con la información correspondiente a las prendas que se encuentran a la venta. De cada prenda se registra: `cod_prenda`, descripción, colores, `tipo_prenda`, stock y `precio_unitario`. Ante un eventual cambio de temporada, se deben actualizar las prendas a la venta. Para ello reciben un archivo conteniendo: `cod_prenda` de las prendas que quedarán obsoletas. Deberá implementar un procedimiento que reciba ambos archivos y realice la baja lógica de las prendas, para ello deberá modificar el stock de la prenda correspondiente a valor negativo.

Por último, una vez finalizadas las bajas lógicas, deberá efectivizar las mismas compactando el archivo. Para ello no podrá utilizar ninguna estructura auxiliar, debe resolverlo dentro del mismo archivo. Solo deben quedar en el archivo las prendas que no fueron borradas, una vez realizadas todas las bajas físicas.

6)

1. Archivos Secuenciales

Una empresa que comercializa fármacos recibe de cada una de sus 30 sucursales un resumen mensual de las ventas y desea analizar la información para la toma de futuras decisiones.

El formato de los archivos que recibe la empresa es: `cod_farmaco`, nombre, fecha, `cantidad_vendida`, `forma_pago` (campo String indicando contado o tarjeta).

Los archivos de ventas están ordenados por: `cod_farmaco` y fecha.

Cada sucursal puede vender cero, uno o más veces determinado fármaco el mismo día, y la forma de pago podría variar en cada venta. Realizar los siguientes procedimientos:

a) Recibe los 30 archivos de ventas e informa por pantalla el fármaco con mayor `cantidad_vendida`.

NO → Recibe los 30 archivos de ventas e informa por pantalla la fecha en la que se produjeron más ventas al contado mostrando fecha y cantidad de pagos contado.

c) Recibe los 30 archivos de ventas y guarda en un archivo de texto un resumen de ventas por fecha y fármaco con el siguiente formato: `cod_farmaco`, nombre, fecha, `cantidad_total_vendida`. (el archivo de texto deberá estar organizado de manera tal que al tener que utilizarlo pueda recorrer el archivo realizando la menor cantidad de lecturas posibles). **Nota:** en el archivo de texto por fecha, cada fármaco aparecerá a lo sumo una vez. Además de escribir cada procedimiento deberá declarar las estructuras de datos utilizadas.