

## ALGORITMOS Y ESTRUCTURA DE DATOS

### ARCHIVOS BINARIOS

- 1) Se dispone de un conjunto de inscripciones de alumnos a examen en el mes de mayo. Cada inscripción tiene los siguientes datos: nombre y apellido, número de legajo, código de materia, día, mes y año del examen. Los datos finalizan con un número de legajo igual a cero.  
Desarrollar un programa que a partir del ingreso de las inscripciones por teclado, genere un archivo binario de inscripción de alumnos a exámenes finales DIAFINALES.DAT, según el siguiente diseño:  
a.1 Nro. de legajo                      a.2 Código de materia  
a.3 Día del examen                      a.4 Mes del examen  
a.5 Año del examen                      a.6 Nombre-Apellido
- 2) Dado el archivo binario generado en el ejercicio 1, desarrolle un programa que solicitando por teclado un código de materia permita seleccionar todos los registros de alumnos que se anotaron para rendirla y los grave en otro archivo (MATFINALES.DAT), con el mismo diseño.
- 3) Dado el archivo binario generado en el ejercicio 1, que contiene todas las inscripciones del día, y otro con el mismo diseño que contiene las inscripciones anteriores (FINALES.DAT), desarrolle un programa que agregue al archivo de inscripciones anteriores el contenido del archivo del día.
- 4) Dado un archivo binario de productos que contiene 100 registros, y cada registro corresponde a un producto que está codificado del 1 a 100, ordenado por código de producto con el siguiente diseño:  
código de producto (1..100)                      stock  
y otro archivo binario de pedidos, con el siguiente diseño:  
número de pedido                      número de cliente  
código de producto (1..100)                      cantidad pedida

Se pide desarrollar un algoritmo que:

- a) Realice un listado con los pedidos que fueron satisfechos en su totalidad según el siguiente listado:

| Pedidos Satisfechos |                   |                    |                 |
|---------------------|-------------------|--------------------|-----------------|
| Número de pedido    | Número de Cliente | Código de Producto | Cantidad pedida |
| 9999                | 999999            | 999                | 9999            |
| 9999                | 999999            | 999                | 9999            |

- b) actualizar el campo stock del archivo de producto, cada vez que acepte un pedido.
- c) grabar un archivo binario con el mismo diseño que el archivo de pedidos, con aquellos pedidos que no pueden ser satisfechos en su totalidad.

### Apareo, Corte de Control, y altas, bajas y modificaciones en archivos binarios.

- 5) Se dispone un archivo binario de inscripción de alumnos a exámenes finales MAESTROFINALES.DAT y otro con las inscripciones del día de hoy DIAFINALES.DAT, ambos ordenados ascendente por número de legajo y código materia y con el siguiente diseño:

a.1 Nro de legajo                      a.2 Código de materia                      a.3 ApellidoNombre

Se pide desarrollar un programa que genere un nuevo archivo de inscripciones a finales FINALESACT.DAT resultante del apareo de los dos archivos anteriores, con el mismo orden y diseño.

- 6) Una empresa de canales por cable desea actualizar el archivo de series que emite por sus distintos canales, con material nuevo. Para ello posee los siguientes archivos:

- a) un archivo maestro de series, **Series.dat**, con un registro con los datos de las series, *ordenado ascendente por Id\_Serie*, con el siguiente diseño:

a.1) Id\_Serie                      a.2) Título de la serie                      a.3) Genero

- b) otro archivo de novedades de series, **NovSeries.dat**, con el mismo diseño y orden que el archivo anterior, que contiene las novedades a incorporar.

Se pide desarrollar la metodología necesaria para escribir un algoritmo que Grabe un archivo maestro de series actualizado, **ActSeries.dat**, con el mismo diseño y orden que los anteriores.

- 7) El dueño de un local de venta de libros desea relevar el stock que posee en el local y en el depósito, para realizar las compras del mes.

Para ello cuenta con dos archivos:

- a) **StockEnLocal.dat**, *ordenado por código del libro*, con un registro por cada libro que se encuentra en el local, con el siguiente diseño:

a.1) Código de libro                      a.2) Título del libro (30 caracteres)  
a.3) Autor (20 caracteres)                      a.4) Editorial (20 caracteres)  
a.6) Stock en el local                      a.7) Genero ( 10 caracteres)

- b) **StockEnDeposito.dat**, *ordenado por código del libro*, con un registro por cada libro que se encuentra en el depósito, con el siguiente diseño:

b.1) Código de libro                      b.2) Stock en depósito

Se pide desarrollar la metodología necesaria para escribir un algoritmo que emita un listado *ordenado por código de libro*, con un renglón por cada libro que tenga faltante en stock sea en depósito, local o en ambos lugares, con el siguiente formato:

| Libros faltantes              |                              |
|-------------------------------|------------------------------|
| Código                        | Observación                  |
| 9999                          | Falta en depósito            |
| 9999                          | Falta en local               |
| 9999                          | Falta en local y en depósito |
| Total de libros en falta: 999 |                              |

- 8) Una aplicación para descargas de Series por la red posee la información en un archivo binario, **Episodios.dat**, con un registro por cada episodio, *ordenado ascendente por IdSerie y Número de temporada*, con el siguiente diseño:

1) IdSerie                      2) Título del episodio (20 caracteres)  
3) Número de temporada                      4) Número de episodio  
5) Cantidad de descargas                      6) Fecha de última descarga (aaaammdd)

Se pide desarrollar la metodología necesaria para escribir un algoritmo emita el siguiente listado:

# Listado de Descargas de Series

Serie: 999999999

Temporada: 99

| N. de Episodio | Título del Episodio | Cant. descargas | Fecha de última descarga |
|----------------|---------------------|-----------------|--------------------------|
| 999            | xxxxxxxxxxxxxx      | 9999999         | dd/mm/aaaa               |
| 999            | xxxxxxxxxxxxxx      | 9999999         | dd/mm/aaaa               |

\*Cant. Total de Episodios de la Temporada: 99999

\*Total descargas de la temporada:999999999

Serie: 999999999

Temporada: 99

| N. de Episodio | Título del Episodio | Cant. descargas | Fecha de última descarga |
|----------------|---------------------|-----------------|--------------------------|
| 999            | xxxxxxxxxxxxxx      | 9999999         | dd/mm/aaaa               |
| 999            | xxxxxxxxxxxxxx      | 9999999         | dd/mm/aaaa               |

\*Cant. Total de Episodios de la temporada: 99999

\*Total descargas de la temporada: 999999999

\*\*Cant. Total de Episodios de la Serie: 99999

\*\*Total descargas de la Serie: 999999999

\*\*\*Total General de series: 9999

- 9) El dueño de un local de venta de juegos para distintas consolas necesita desarrollar un algoritmo que grabe un archivo, **JuegosPorConsola.dat, ordenado por consola**, con un solo registro por consola según el siguiente diseño:

a) Consola (10 caracteres)      b) Cantidad de juegos

Para obtener la información solicitada se cuenta con el archivo **Juegos.dat, ordenado por consola**, con un registro por cada juego que se encuentra en el local, con el siguiente diseño:

|                      |                                     |
|----------------------|-------------------------------------|
| 1) Código del juego  | 2) Titulo del Juego (30 caracteres) |
| 3) Stock en el local | 4) Consola (10 caracteres)          |

- 10) Dado el archivo ‘ALUMNOS.dat’ con los datos personales de alumnos ordenado por legajo, con el siguiente diseño:

|                               |  |
|-------------------------------|--|
| a.1 Legajo                    | a.2 Apellido y nombre ( 30 caracteres) |
| a.3 Domicilio (20 caracteres) | a.4 Código postal                      |
| a.5 Teléfono (10 caracteres)  | a.6 Año de ingreso                     |

Y otro archivo con el mismo orden que el mencionado llamado “NOVEDADES.dat”, con las actualizaciones (altas, bajas, y modificaciones) a ser aplicadas, donde cada registro contiene además de todos los campos de Alumnos.dat un código de operación (‘A’: Alta, ‘B’: Baja, ‘M’: Modificación).

Se pide desarrollar todos los pasos necesarios para realizar un programa que genere un archivo actualizado “ALUMACTU.dat” con el mismo diseño.