



5. EJERCICIOS

- 1.- Dado un fichero binario conteniendo datos enteros (Z), elaborar un algoritmo que escriba en un fichero de texto los valores ordenados del fichero de partida.
- 2.- Elaborar un procedimiento para concatenar dos ficheros binarios con el mismo tipo de registro. El resultado deberá ser otro fichero conteniendo los datos del primer fichero seguidos de los datos del segundo fichero.
- 3.- Diseñar un algoritmo que recoja del teclado un valor numérico entero y el nombre de un fichero binario conteniendo datos enteros; y que busque en dicho fichero el valor dado. En caso de encontrarlo el algoritmo deberá indicar el lugar (posición del registro) en el que se ha encontrado por primera vez.
- 4.- Dado un fichero de texto, constituido únicamente por cadenas que representen valores enteros, elaborar un algoritmo que escriba en otro fichero los valores ordenados del fichero de partida.
- 5.- Elaborar un algoritmo que lea un fichero de texto en disco y cuente el número de caracteres, el número de palabras y el número de líneas que contiene el mismo, presentado estos datos como resultado en pantalla.

Para contar las palabras, considerar como una palabra cualquier sucesión de caracteres separada por uno o más blancos, tabuladores, símbolos de puntuación o fin de línea.

- 6.- Elaborar un algoritmo que calcule el porcentaje de números capicuas que existan en un fichero binario conteniendo datos enteros.
- 7.- Elaborar un algoritmo que presente en pantalla un fichero de texto cualquiera.
- 8.- Dada una palabra, diremos que está “escondida” en un texto si está contenida en éste, no necesariamente de forma continua. Por ejemplo, la palabra “elefante” está escondida en la frase “El

algoritmo tal que, dada una palabra leída por teclado, la busque en un texto almacenado en un fichero de texto. Como resultado el algoritmo deberá presentar el mensaje:

```
Palabra <palabra> no encontrada en el texto.
```

O bien

```
Palabra <palabra> encontrada escondida entre los <p> primeros caracteres del texto.
```

En el ejemplo el resultado debería ser:

```
Palabra elefante encontrada escondida entre los 32 primeros caracteres del texto.
```

- 9.- Se define el tipo:

Tipos

```
fecha = REGISTRO DE
      día : [1..31]
      mes : (enero, febrero, marzo, abril, mayo, junio, julio,
            agosto, septiembre, octubre, noviembre, diciembre)
      año : [1600..2000]
FINREGISTRO
```

Se desea ordenar el fichero binario *fecha.dat* cuyas componentes son del tipo *fecha*. Elaborar un algoritmo para ordenar el fichero *fecha.dat* y descargar en el fichero de texto *ordenado.txt* las fechas ordenadas a razón de una por línea.

- 10.- De los distintos hospitales de la ciudad de Boston se han recogido datos a fin de estudiar el efecto que producen diversos tratamientos sobre ciertas enfermedades cancerígenas. Supuestas las siguientes definiciones de tipo:

Tipos

```
hospital = ( HC, BosH, MassHC, Univ, USS )
enfermedad = ( leucemia, piel, estomago, pulmon, laringe )
tratamiento = [1..6]
bloque = ARRAY hospital, enfermedad, tratamiento DE N
```

Se han estructurado los datos en una matriz de tipo "bloque", de modo que en la componente de *i, j, k* se encuentra el número de enfermos curados en el hospital *i*, de la enfermedad *j*, mediante el tratamiento *k*. Esta matriz está grabada como primera componente del fichero binario ESTUDIO.DAT que contiene componentes del tipo "bloque".

Elaborar un algoritmo que presente en pantalla la siguiente información:

- (i) Número total de enfermos curados de la enfermedad *j* mediante el tratamiento *k*. Debe aparecer un listado que recoja todas las posibles combinaciones de la enfermedad y tratamiento; cada una de sus líneas debe ser de la forma:

< ENFERMEDAD > < TRATAMIENTO > < NUMERO DE CURADOS >

- (ii) Número total de enfermos curados en cada uno de los hospitales. Debe darse una presentación similar a la pedida en el apartado (i).

- 11.- Dado un fichero de texto, elaborar un algoritmo que efectue el recuento del número de líneas de dicho fichero que tienen cero caracteres, un carácter, dos, tres, etc. y presente por pantalla un listado que recoja estos resultados. Sólo debe aparecer en el listado el número de líneas con *j* caracteres si hay al menos una línea con *j* caracteres en el fichero.

Ejemplo: Si el contenido del fichero fuera:

```
FECHA DEL EXAMEN :<EOL>
<EOL>
dia tres<EOL>
de junio<EOF>
```

el listado debería ser:

```
Caracteres : 0   Líneas : 1
Caracteres : 8   Líneas : 2
Caracteres : 17  Líneas : 1
```

Nota: Se considerará que la longitud máxima de una línea es de 255 caracteres.

- 12.- Se desea gestionar un listín telefónico, de tal manera que cada entrada del listín contenga:

Nombre de la persona.
Dirección.

Para ello deberá definirse un tipo de fichero para almacenar los datos del listín y diseñar un algoritmo que permita:

1. Añadir una entrada al listín.
2. Consultar una entrada del listín dado el número de teléfono.
3. Modificar un dato dado el número de teléfono.
4. Borrar una entrada del listín dado el número.

- 13.- Dado un fichero secuencial de nombre `fichero_ini` y una propiedad `P`, crear dos ficheros secuenciales `fichero_P` y `fichero_NoP` con los registros del primero que satisfacen y no satisfacen la propiedad `P` respectivamente.
- 14.- Se quiere llevar el control de los subscriptores de una revista. Para esto se utiliza un fichero binario de acceso secuencial de nombre `subscrip` que contiene los datos de aquellos, en registros del tipo:

Tipos

```
TipoCadena = ARRAY [1..80] DE $
TipoCadenaCorta = ARRAY [1..15] DE $
TipoEstudios = (Sin, Basico, Bachiller, Gmedio, Superior)
TipoSubscriptor = REGISTRO DE
    Num : N
    Nombre : TipoCadena
    Direccion : TipoCadena
    Ciudad : TipoCadenaCorta
    Estudios : TipoEstudios
    Telefono : TipoCadenaCorta
FINREGISTRO
```

- (1) Diseñar un algoritmo para crear el fichero `subscrip` leyendo los datos desde el teclado hasta leer un dato con el campo `Num` igual a cero.
- (2) Diseñar un algoritmo para añadir nuevas subscripciones.
- (3) A partir del fichero de subscripciones anterior y bajo la hipótesis de que hay al menos un subscriptor y que ningún subscriptor aparece dos veces, escribir un algoritmo que permita modificar los datos de un subscriptor dado.