

Unit DirExt;

{Objetivo: Primitivas de manejo de un archivo directo extensible. Los elementos pueden ser de cualquier tipo: tipos predefinidos de Pascal, registros de datos completos o registros de indice a registros de datos (clave de identificacion de registro de datos con referencia a registro). En cualquier caso se especifica cual es la clave de identificacion de los elementos en un tipo exclusivo y en una funcion primitiva (de interfaz) que la devuelve.

Autor: Lic. Arturo Servetto

Nota: los comentarios estan escritos obviando acentos y otros signos diacriticos para evitar problemas de compatibilidad.}

Interface**Const**

*TB=16; {Tamano de bloque (debe ser un valor que sea submultiplo o multiplo del tamano de los registros fisicos de los dispositivos de almacenamiento persistente o de las unidades de asignacion de espacio a archivos del sistema de archivo del sistema operativo a utilizar: 2^n*512, con n desde -5 para nodos de 16 bytes)}*

Type

TElemento=Word; {tipo de elementos a almacenar en el archivo (en este caso numeros naturales entre 0 y 65535)}
TClave=Word; {tipo de clave de identificacion de elementos (en este caso la clave es el mismo elemento, por tratarse de elementos simples)}
TRefB=Byte; {tipo de referencias a bloques}

Const

```
CE=(TB-3) div SizeOf(TElemento);  {cantidad maxima de
elementos en un bloque}
CR=(TB-3) mod SizeOf(TElemento);  {cantidad de bytes
de relleno}
```

Type

```
{$PackRecords 1} {directiva al compilador (solo para
Free Pascal) para
que almacene los campos de registros direccionando de
a byte}
```

```
TBloque=Record {nodo de arbol B}
    cont: Byte;  {contador de elementos contenidos
efectivamente}
    td: Word; {tamano de dispersion del bloque}
    el: Array[1..CE] of TElemento;
    relleno: Array[1..CR] of Byte;  {relleno para
completar el tamano
del bloque}
end;
```

```
TTabla=File of TRefB;
```

```
TArchDE=Record {archivo organizado con todos los
archivos y datos de
control necesarios para abrirlo}
    nombre: String;
    datos: File of TBloque; {archivo de nodos}
    tabla: TTabla; {tabla de dispersion}
    libres: File of TRefB;  {archivo de
referencias a bloques libres}
    tamTabla: Word {tamano de la tabla}
end;
```

```
TRes=(ok, duplicado, inexistente);  {resultados
posibles de una insercion,
busqueda o supresion}
```

```
Procedure Crear(var arch: TArchDE);  {crea un archivo vacio
con el bloque 0 vacio
e inicializa la tabla con un unico registro apuntandolo}
```

Procedure Abrir(**var** arch: TArchDE);

Procedure Cerrar(**var** arch: TArchDE);

Function Clave(elem: TElemento): TClave; *{devuelve la clave de un elemento (se debe implementar segun el tipo TElemento)}*

Procedure Insertar(**var** arch: TArchDE; elem: TElemento; **var** result: TRes);
{inserta un elemento excepto que ya exista uno con la misma clave de identificacion, en cuyo caso result vuelve con valor "duplicado"}

Procedure Buscar(**var** arch: TArchDE; claveE: TClave; **var** result: TRes);

Procedure Suprimir(**var** arch: TArchDE; claveElem: TClave; **var** result: TRes);
{elimina el elemento identificado por claveElem o devuelve result=inexistente en caso de que no exista un elemento con esa clave}

Procedure Exportar(**var** arch: TArchDE; nom: **String**);*{exporta el archivo (la tabla y los datos) a un archivo de texto}*