

Estructuras de datos: Archivos

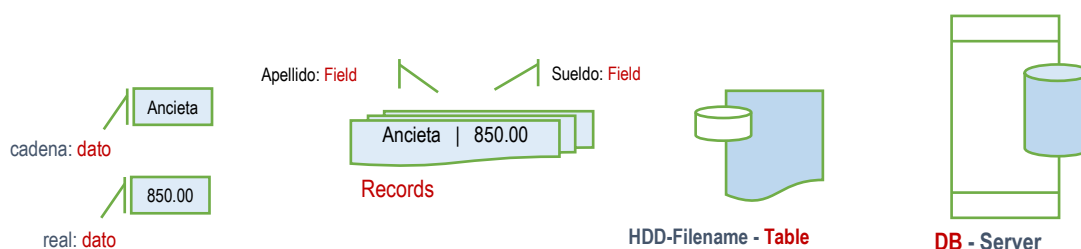
Archivos secuenciales Ejemplos comentados

3. Definición de Fichero o archivo.

Las estructuras de datos, aplicadas al almacenamiento secundario de colecciones de datos se denominan "**Organización de Archivos**". Un fichero por tanto es. "**Una estructura de datos, cuyos elementos básicos, los campos que pertenecen a un tipo de datos, se organizan linealmente en registros de un mismo tamaño, y estos a su vez en tablas de registros de longitud variable.**"

Jerarquía de la Organización de Archivos

- **Datos**
- **Campos.**
- **Registros.**
- **Ficheros**
- **Base de Datos**



1Fig. Jerarquía de la Organización de Archivos

5. Procesamiento de archivos secuenciales.

En un archivo secuencial los datos se insertan en el dispositivo de almacenamiento en orden cronológico de llegada. (Inmediatamente a continuación del anterior).

Los archivos secuenciales terminan con una marca de "Final De Archivo: FDA" (End Of File: EOF en ingles). Cuando se agregan nuevos datos se añaden (append) antes de las marcas EOF.

Por contraparte, un Archivo secuencial también posee una marca de "Inicio De Archivo: IDA" (Begin Of File: BOF en ingles).

area BOF
1. registro
2. registro
...
n. registro
area EOF

Fig. 1. Estructura de un archivo secuencial (s).

Problemas resueltos

Problema 1. Algoritmo de acceso secuencial: Crear archivo para almacenar caracteres.

Análisis:

El dispositivo a escribir corresponde a una estructura de tipo carácter

Especificaciones de E/S

Entrada: caracteres

Salida: dispositivo de disco de tipo carácter

Pseudocódigo

Algoritmo Escribir caracteres en disco.

tipo

archivo_s de caracter: identificador_fc

var

 identifiacodr_fc: fc // canal de comunicacion

 caracter: letra

inicio

 letra ← 'a' //inicializar var

 //paso1: definir dispositivo

crear(fc, "datos1.txt")

abrir(fc, e, "datos1.txt")

 //validar definición de archivo

si (fc = NULL) **entonces**

escribir("Error: no se puede abrir archivo")

si_no

 //paso2: escribir caracter

escribirArchivo(fc, letra)

fin_si

 //paso 3: cerrar canal de comunicacion

cerrar(fc)

 //explore el sistema de archivos de la PC y verifique el archivo

fin



Problema 2. Algoritmo de acceso secuencial: Leer archivo que almacena caracteres.

Análisis:

El dispositivo a leer corresponde a una estructura de tipo carácter

Especificaciones de E/S

Entrada: dispositivo de disco de tipo carácter
Salida: caracteres

Pseudocodigo

Algoritmo leer caracteres desde disco.

tipo

archivo_s de caracter: identificador_fc

var

identifiacodr_fc: fc // canal de comunicacion
caracter: letra

inicio

//paso1: definir dispositivo

abrir(fc, l, 'datos1.txt') //l- modo lectura

//validar identificador de archivo

si (fc = NULL) **entonces**

escribir('Error: no se puede abrir archivo')

si_no

//paso2: leer caracter

letra ← **leerArchivo**(fc)

escribir(letra)

fin_si

//paso 3: cerrar canal de comunicacion

cerrar(fc)

//explore el sistema de archivos de la PC y verifique.

fin

Comprobación de algoritmos: Implementación C++

2. Código C++: Modo Source File

```
/*Name: File_EscribirChar.cpp
  Author: J.Medianero.A
  Date: 25/09/08 09:42
  Description: Define archivo en modo escritura de tipo caracter
*/

#include <cstdlib>
#include <iostream>

using namespace std;

int main(int argc, char* argv[]){
    FILE *pNumFile;
    char letra = 'a';

    pNumFile = fopen("Letras.txt", "w"); //crea un fichero de escritura
                                         //si ya existe, lo crea de nuevo
    //pNumFile = fopen("Letras.txt", "a"); //abre o crea un fichero para
                                         //añadir datos al final del mismo

    if( pNumFile == NULL ) {
        printf("Error de acceso");
        system("PAUSE");
        exit(0); //abandonar aplicacion
    }else{
        fputc(letra, pNumFile);
        fclose(pNumFile);
        //system("NotePad Letras.txt");
    }

    system("PAUSE");
    return EXIT_SUCCESS;
} //fin main
```



```
/*Name: File_LeerChar_v1.cpp
  Author: J.Medianero.A
  Date: 25/09/08 09:42
  Description: Define archivo en modo lectura de tipo caracter
*/
#include <cstdlib>
#include <iostream>

using namespace std;

int main(int argc, char* argv[]){
    FILE *pNumFile;
    char letra;

    pNumFile = fopen("Letras.txt", "r");    //abre un fichero en modo lectura
                                           //si no existe retorna NULL

    if( pNumFile == NULL ) {
        printf("Error de acceso...");
        cin.get();
        exit(0); //abandonar aplicacion
    }else{
        letra = fgetc(pNumFile);
        printf("%c", letra);
    }
    fclose(pNumFile);
    //system("Notepad Letras.txt");
    system("PAUSE");
    return EXIT_SUCCESS;
} //fin main
```

```
/*Name: File_LeerChar_v2.cpp
  Author: J.Medianero.A
  Date: 02/01/10 09:42
  Description: Define archivo en modo lectura de tipo carácter.
  NOTA: Si la escritura de archivo esta definido en modo "a" (append), los
  datos se acumulan en el archivo. Para leer todos los datos guardados, debe
  modificar la rutina de lectura, implementando un bucle controlado por
  agotamiento (hasta encontrar el final del archivo -EOF: End Of File-)
*/
//define prototipo: leerArchivoDisco
procedimiento leerArchivoDisco(cadena: nombreArchivo)
var
    identificador_fc: fc
    caracter: letra
inicio
    abir(fc, 1, nombreArchivo)
    si (fc = null) entonces
        escribir("Error de acceso")
    si_no
        repetir
            letra ← leerArchivo(fc)
            escribir(letra)
        hasta_que( EOF )
    fin_si
    cerrar(fc) //cerrar canal de comunicacion
fin
```