

UF1.1

Ficheros

PEDRO J. CAMACHO



**Universidad
Europea de Madrid**

LAUREATE INTERNATIONAL UNIVERSITIES

CONTENIDOS

1. Aprender qué son y cómo se utilizan los archivos secuenciales , aleatorios e indexados.
2. Entenderlos en un contexto de evolución histórica.
3. Conocer sus ventajas e inconvenientes.
4. Entender de qué manera han influido en el posterior diseño de las bases de datos relacionales



CASO: LAS MEJORES PUNTUACIONES DE UN JUEGO

¿Cómo lo harías?

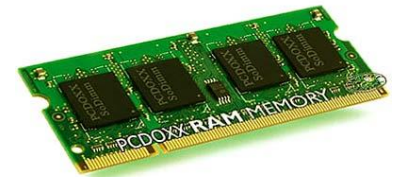
¿Dónde se guardan las puntuaciones de un juego?

- Aunque apaguemos el ordenador/console, al volver a jugar después, aparecen los récords anteriores, ¿por qué?

¿Qué es la memoria volátil?

¿Y el almacenamiento secundario?

Gran dependencia de la tecnología.



VOLVAMOS ATRÁS

¿Qué pasaba hace 30 años?

¿Qué información almacena una empresa?

- A quién vende -> Clientes
- A quién compra -> Proveedores

¿Qué ventajas tenía entonces utilizar ordenadores?

- Búsquedas más rápidas
- Contabilidad
- Facturación



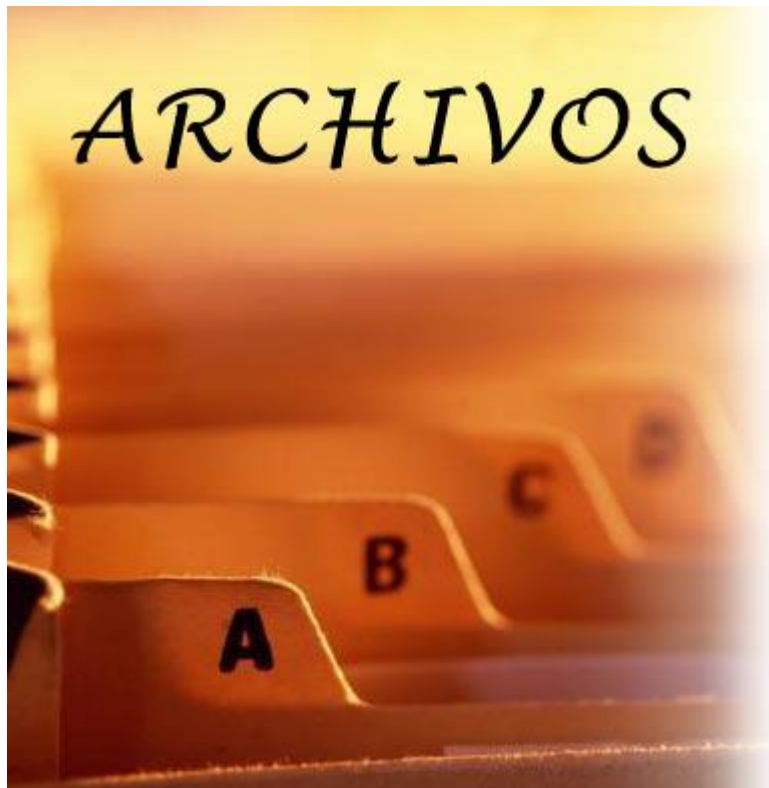
¿Archivos o ficheros?

La mayor parte de aquellas empresas utilizaban archivos de papel, donde guardaban datos de sus clientes en fichas escritas con máquina de escribir. Por eso los archivos reciben también el nombre de ficheros



Dependiendo del método de acceso a los datos

- Secuenciales
- Acceso Aleatorio
- Indexados



ARCHIVOS SECUENCIALES

Volvamos de nuevo atrás

En casa de nuestros padres es posible que encontremos cintas de música llamadas cassettes:



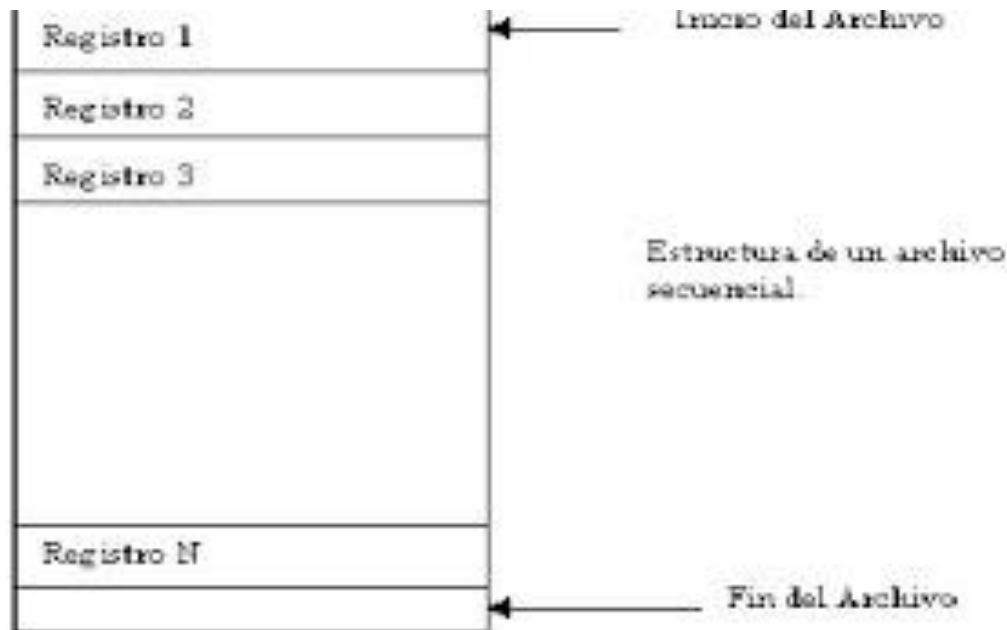
Las unidades de cintas informáticas eran como unos cassettes gigantes donde se almacenaban datos digitales: ceros y unos de forma secuencial.



ARCHIVOS SECUENCIALES

Volvamos de nuevo atrás

En aquel sistema de almacenamiento, lo que se podía hacer era crear un fichero de clientes donde se escribía el nombre, dirección, población, etc. (que se llama campo). Todos ellos separados por retornos de carro (enter). Entre cliente y cliente se escribía una marca de fin de bloque. Cada uno de estos bloques de datos se llama registro.





ARCHIVOS SECUENCIALES

Un ejemplo de escritura – Tarea 1



Ve al campus virtual y en la sección de Recursos descárgate el fichero comprimido “Tarea 1 – C#” que contiene dos archivos:

- Program1A.cs (código fuente en C#)
- Actividad1A.exe (código ejecutable)

Ejecuta (doble clic) el fichero “Actividad1A.exe”

Abre el nuevo fichero que se ha creado y observa su estructura.

```
CLIENTES.TXT: Bloc de notas
Archivo  Edición  Formato  Ver  Ayuda
Juan Martínez
Calle del Pez, 5
Madrid
<FIN>
Comercial Martínez
Calle de la Cuesta, 10
Sevilla
<FIN>
José Sánchez
Calle Mayor, 7
Salamanca
<FIN>
```



ARCHIVOS SECUENCIALES

Un ejemplo de escritura – Tarea 1



Ahora vamos a ver su código fuente, para ello edita (blog de notas) el fichero de código fuente, ¿entiendes lo que pone?, ¿en qué formato está almacenado?

```
using System.Collections.Generic;
using System.Text;
using System.IO;

namespace Actividad1D
{
    class Program
    {
        static void Main(string[] args)
        {
            StreamWriter archivo = null;

            try
            {
                Console.WriteLine("Ejemplo de escritura de 3 registros en un archivo secuencial");

                archivo = new StreamWriter("CLIENTES.TXT");
                if (archivo != null)
                {
                    //Cliente 1:
                    archivo.WriteLine("Juan Martínez");
                    archivo.WriteLine("Calle del Pez, 5");
                    archivo.WriteLine("Madrid");
                    archivo.WriteLine("<FIN>");
                    //Cliente 2:
                    archivo.WriteLine("Comercial Martínez");
                    archivo.WriteLine("Calle de la Cuesta, 10");
                    archivo.WriteLine("Sevilla");
                    archivo.WriteLine("<FIN>");
                    //Cliente 3:
                    archivo.WriteLine("José Sánchez");
                    archivo.WriteLine("Calle Mayor, 7");
                    archivo.WriteLine("Salamanca");
                    archivo.WriteLine("<FIN>");
                    archivo.WriteLine();
                    //Cerrar el archivo:
                    archivo.Close();
                    Console.WriteLine("3 registros grabados.");
                }
            }
            catch (IOException e)
            {
                Console.WriteLine("Error: " + e.ToString());
            }
        }
    }
}
```



TAREA 1

Escritura Ficheros Secuenciales

Con lo que has aprendido en clase, prueba a hacerlo tú y sube la actividad al campus virtual





ARCHIVOS SECUENCIALES

Características

- No permite el retroceso (forward only)
- Los archivos secuenciales son monousuarios
- Estructura rígida de campos (en orden)
- El modo de apertura condiciona lectura o escritura
- Lecturas parciales pero escrituras totales
- La marca de final de archivo (EOF)
- Borrado de registros omitiendo contenido
- La posibilidad de uso de la marca de sincronismo (<FIN>)
- Registros de longitud variable
- Contenido legible en un procesador de textos





ARCHIVOS SECUENCIALES

Consideraciones finales



- Siguen siendo una buena opción para almacenamiento de pocos registros, donde la velocidad de acceso no sea crítica.
- Los Sistemas Operativos, soportan I/O sobre ellos
- Todos los Lenguajes de Programación tienen funciones para acceder a ellos



ARCHIVOS SECUENCIALES

Un ejemplo de lectura – Tarea 2



Ve al campus virtual y en la sección de Recursos descárgate el fichero comprimido “Tarea 2 – CPP” que contiene tres archivos:

- Program2A.cpp (código fuente en C++)
- Actividad2A.exe (código ejecutable)
- CLIENTES.txt

Ahora abre una consola (cmd) desplázate hasta el directorio donde has descomprimido el fichero y ejecuta el “exe”. ¿Qué pasa?

Edita el fichero de clientes, añade un nuevo registro y vuelve a ejecutar el programa.



ARCHIVOS SECUENCIALES

Un ejemplo de lectura – Tarea 2



Ahora vamos a ver su código fuente, para ello edita (blog de notas) el fichero de código fuente, ¿entiendes lo que pone?, ¿en qué formato está almacenado?

```
Actividad2A.cpp: Bloc de notas
Archivo Edición Formato Ver Ayuda
// Actividad3A.cpp : Defines the entry point for the console application.
//
#include "stdafx.h"
#include <string.h>
//Actividad 3A:
//Lectura de archivo secuencial con tres clientes.

int main(int argc, char* argv[])
{
    FILE * pArchivo = NULL;
    char szLinea[255];

    pArchivo = fopen("CLIENTES.TXT", "r");
    if (pArchivo)
    {
        while (!feof(pArchivo))
        {
            memset(szLinea, 0, sizeof(szLinea));
            fgets(szLinea, 255, pArchivo);
            //Eliminar último carácter LF:
            *(szLinea + strlen(szLinea)-1)=0;
            printf("%s\n", szLinea);
        }
        fclose(pArchivo);
    }

    return 0;
}
```

TAREA 2

Lectura Ficheros Secuenciales

Con lo que has aprendido, prueba a hacerlo tú y sube la actividad al campus virtual

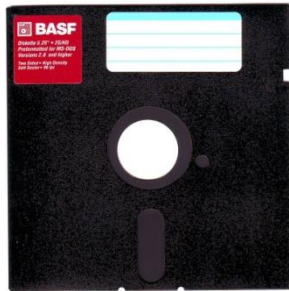




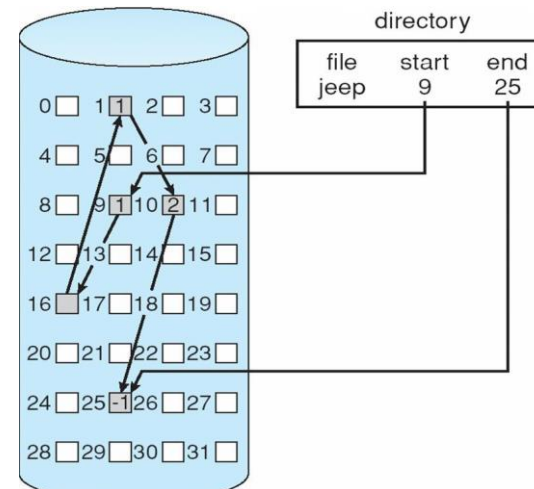
ARCHIVOS DE ACCESO ALEATORIO

Volvamos de nuevo atrás

Los archivos de acceso aleatorio aparecen con el disquete y el disco duro:



En ellos podemos situarnos en una posición determinada sin pasar por el inicio, gracias a su número de registro.





ARCHIVOS ALEATORIOS

Características



- Posicionamiento inmediato
- Registros de longitud fija
- Apertura para lectura/escritura
- Permiten el uso concurrente (multiusuario)
- Dimensionamiento máximo al ser creados
- Borrado de registro mediante ceros



ARCHIVOS SECUENCIALES

Consideraciones finales



- Los archivos de acceso aleatorio están especialmente indicados para aquellos casos en que el código del registro (cliente, proveedor, etc.) pueda ser directamente el número de registro en el que se guardan los datos.
- Al trabajar con archivos de acceso aleatorio deberemos tener en cuenta no rebasar nunca el final de archivo.
- Los archivos de acceso aleatorio son un paso más que nos conduce por el camino de los archivos indexados, puesto que constituyen el depósito de los datos. Como veremos más adelante, el almacenamiento temporal o no de las claves es el elemento que lo diferencia de los indexados.

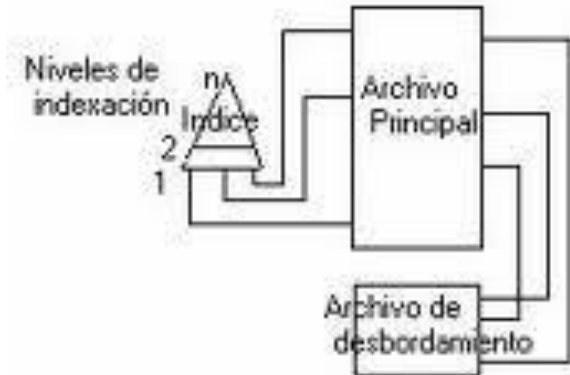
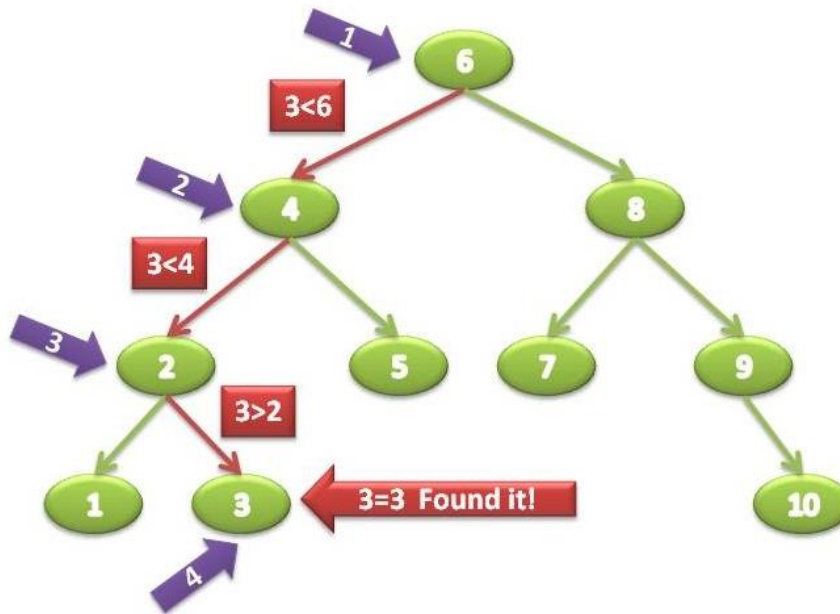
ARCHIVOS INDEXADOS

¿Qué son?

Son básicamente archivos de acceso aleatorio, a los que se añade una estructura de índice, en la que se utiliza una clave.

En el caso de los archivos de acceso aleatorio, la clave era siempre el número de registro.

Se basan en los árboles binarios equilibrados.





RESUMEN

Recapitulemos...

- Los antecesores de las bases de datos son los archivos de almacenamiento de datos.
- Históricamente se ha pasado por tres fases, que corresponde a los tres tipos de archivos visto: secuenciales, aleatorios e indexados.
- La tecnología de los indexados está presente en los SGBD.
- Podemos escribir programas en diferentes lenguajes, que trabajan con estos tres tipos de ficheros, que siguen siendo una buena opción para tareas sencillas y robustas.



RESUMEN



ALMACENAMIENTO EN LA NUBE

Tutorial (6:35)

