

Fundamentos de Organización de Datos

1

Curso 2015

La cátedra

Profesores:

Mg. Rodolfo Bertone Turno 1

Mg. Thomas Pablo Turno 2

Lic. Marrero Luciano Turno1 / Turno 2

Trabajos Prácticos:

JTP: Lic. Serveto Arturo Turno 1

JTP: Lic. Sobrado Ariel Turno 2

JTP: Lic. Restelli Noelia Turno 1 / Turno 2

La cátedra

- **Clases**
 - Teóricas
 - Explicaciones de Prácticas (donde se presentan ejemplos)
 - Prácticas
 - Se utilizará la plataforma WebUnlp - Curso IBBDD / FOD
- **Para aprobar la cursada**
 - Un Parcial – 1º fecha Martes 09/06
 - Dos recuperatorios – 2º Fecha Martes 30/06 – 3º Fecha Martes 14/7

Calendario

Semana	Teoría	Práctica
02/03	Introducción. Archivos. Algorítmica Básica	Sin Actividad
09/03	Archivos. Algorítmica Básica. Algorítmica Clásica	Archivos. Algorítmica Básica
16/03	Archivos. Algorítmica Clásica. Proceso de Raíz en Archivos	Archivos. Algorítmica Básica
23/03	Archivos con Registros de Longitud Variable	Sin actividad por feriado
30/03	Semana Santa	Archivos. Algorítmica Clásica
07/04	Rústica de información en Archivos. Índices	Archivos. Algorítmica Clásica
14/04	Arboles Binarios AVL. Introducción a Arboles R	Raias. Registros de Longitud Variable
21/04	Arboles R. Creación. Rústica. Eliminación. Performance	Raias. Registros de Longitud Variable
28/04	Sin actividad por feriado	Arboles
04/05	Arboles R * v R+	Arboles
11/05	Hashing	Arboles
18/05	Hashing	Hashing
25/05	Consulta	Hashing
01/06	Examen de Archivos	Consulta
08/06	Consulta	Primer Parcial (Martes 09/06/2015)
15/06	Examen de Arboles	Consulta
22/06	Consulta	Muestra de examen. consulta

Bibliografia

- Introducción a las Bases de Datos. Conceptos Básicos (Bertone, Thomas)
 - Estructuras de Archivos (Folk-Zoellick)
 - Files & Databases: An Introduction (Smith-Barnes)
 - Fundamentos de Bases de Datos (Korth Silvershatz)

IBD → FOD 2015

- Recursantes de IBD que se pasen a los planes 2015 y, que hayan rendido y aprobado el primer parcial de IBD en el año 2012 o posterior, se les dará aprobada la cursada de FOD.
- Deberán rendir el final de acuerdo al programa de FOD Plan 2015.

Fundamentos de Organización de Datos

Explicación nro 1: **Archivos**

Archivos - Tipos de Archivos

De registros de longitud fija (File of <tipo de registro>)

De bloques de bytes (File): más adelante lo vemos.

De Texto (Text): de caracteres estructurados en líneas → lectura/escritura con conversión automática de tipos (se puede escribir tipos numéricos a archivos de texto, o leer con variables de tipo numérico desde archivos de texto), con acceso exclusivamente secuencial. Estos archivos son muy útiles para cargar archivos de otros tipos con datos editados y validados, y para exportar archivos de otros tipos a un formato accesible por cualquier editor de textos.

Archivos – Acceso a la información

Secuencial: el acceso a cada elemento de datos se realiza luego de haber accedido a su inmediato anterior

Secuencial indizado: el acceso a los elementos de un archivo se realiza teniendo una estructura externa. No tiene en cuenta el orden físico

Directo: se recupera un elemento de datos de un archivo en un solo acceso.

Archivos – Operaciones básicas.

Definición de Archivos

Dos formas:

- **var** archivo_logico: file of tipo_de_dato;
- **Type**
archivo = file of tipo_de_datos;

Var archivo_logico: archivo

Archivos - Ejemplos

Type

```
Persona = record
    DNI: string[8];
    Apellido: String[25];
    Nombre: String[25];
    Direccion : String[25];
    Sexo: char;
end;
```

ArchivoEnteros = file of integer;

ArchivoString = file of string;

ArchivoPersonas = file of Persona;

Var enteros: ArchivoEnteros; texto:ArchivoString; personas: ArchivoPersonas;

Archivos - Operaciones

Correspondencia archivo logico – archivo fisico

assign(nombre_logico, nombre_fisico);

Ejemplo:

Assign(enteros, 'c:\archivos\enteros.dat');

Assign(texto, ' c:\archivos\texto.dat');

Assign(personas, 'c:\archivos\personas.dat');

Archivos - Operaciones

Apertura y creacion de archivos

Crear un archivo:

- *rewrite (nombre_logico);*

Abrir un archivo existente:

- *reset (nombre_logico)*

Ejemplo:

rewrite(enteros);

reset(personas);

Archivos - Operaciones

Cierre de archivos

close (nombre_logico);

Transfiere la información del buffer a disco.

Ejemplo:

close(enteros);

close(personas);

Archivos - Operaciones

Lectura y escritura de archivos

read (nombre_logico, var_dato);

write(nombre_logico, var_dato);

El tipo de dato de var_dato== tipo de datos de los elementos del archivo

Ejemplo:

read(enteros,e); ---> e:integer;

close(personas,p); ---> p:persona;

Archivos – Operaciones adicionales

Control de fin de datos

- `eof(nombre_logico);`

Control de tamaño del archivo

- `filesize(nombre_logico);`

Control de posición de trabajo dentro del archivo

- `Filepos(nombre_logico);`

Ubicación física en alguna posición del archivo

- `Seek(nombre_logico, posicion);`

Archivos - Ejercicio

Programa CrearArchivo;

Type

Persona = record

DNI :string[8]

ApellidoyNombre :string [30];

Direccion :string [40];

Sexo :char;

Salario :real;

end;

Archivo de Personas = file of Persona;

Var

Personas : Archivo de Personas;

Nombre fisico : string[12];

Per : Persona;

Archivos - Ejercicio - Continuación

Begin

```
write( 'Ingrese el nombre del archivo:' );
readln(NombreFisico)
```

{enlace entre el nombre lógico y el nombre físico}
Assign(Personas, NombreFisico);

{apertura del archivo para creación}
rewrite(Personas);

```
{lectura del DNI una persona}
readln(per.DNI);

while (per.DNI <> ' ') do
begin
{lectura del resto de los datos de la persona}
readln(per.ApellidoyNombre);
readln(per.Direccion);
readln(per.Sexo);
readln(per.Salario);

{escritura del registro de la persona en el archivo}
write(Personas, per);

{lectura del DNI de una nueva persona}
readln(per.DNI);
end;

{cierre del archivo}
close(Personas);
End.
```

PREGUNTAS ??