Fundamentos de Organización de Datos
Bienvenidos

La Cátedra

Profesores:

Mg. Rodolfo Bertone

Mg. Thomas Pablo

Trabajos Prácticos:

JTP: Lic. Sobrado Ariel

JTP: Lic. Nucilli Emanuel

JTP: APU Cipollone Juan

Clases

- **Clases**
  - Teóricas
  - Explicaciones de Prácticas (donde se presentan ejemplos)
  - Prácticas
  - Se utilizará plataforma moodle para mensajería y material: **asignaturas.info.unlp.edu.ar**

    Enlace provisorio de material práctico:
    https://drive.google.com/drive/folders/1icJHyH90OSXCu95qWicEb91sGanitZdh?usp=sharing

Temas a ver:

- Persistencia de Datos:
  - Archivos
- Acceso a datos, performance en el acceso:
  - Acceso secuencial indizado (árboles)
  - Acceso Directo (Hashing)

Evaluación

- Se evalúan todos los temas vistos
- Cada tema se aprueba de forma independiente.

Fechas de Examen:

- 1° Fecha Martes 10/6
- 2° Fecha Martes 1/7
- 3° Fecha Martes 15/7

Cambios de turno

Selección de turno en https://fod.info.unlp.edu.ar/
Para los cambios de turnos(solo con certificado laboral o cambio con otro compañero de otro turno) tienen como fecha límite el **Martes 25 de Marzo sin excepción**.

Bibliografía

- 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)

Fundamentos de Organización de Datos

Archivos

Tipos de Archivos

Registros de longitud fija o tipados (File of <tipo_dato>)

Texto(Text): Caracteres estructurados en líneas.
Lectura/escritura con conversión automática de tipos.
El acceso es exclusivamente secuencial.
Útiles para importar y exportar datos.

Bloques de bytes (File)

Operaciones básicas

Definición de Archivos tipados

Dos formas:

- **var** archivo_logico: **file of** tipo_de_dato;

- **type**

  archivo = **file of** tipo_de_datos;

  var archivo_logico: archivo

Ejemplo

```
type
    persona = record
            dni: string[8];
            apellido: string[25];
            nombre: string[25];
            direccion: string[25];
            sexo: char;
    end;
     archivo_enteros  = file of integer;
     archivo_string   = file of string;
     archivo_personas = file of persona;
var
    enteros: archivo_enteros;
    texto: archivo_string;
    personas: archivo_personas;
```

Operaciones

```
assign(nombre_logico, nombre_fisico);
```

Realiza una correspondencia entre el archivo lógico y archivo físico.

Ejemplo:

```
assign(enteros, 'c:\archivos\enteros.dat');
assign(texto,' c:\archivos\texto.dat');
assign(personas, 'c:\archivos\personas.dat');
```

Operaciones

Apertura y creación de archivos

```
rewrite(nombre_logico);
```
→ Crea un archivo

```
reset(nombre_logico);
```
→ Abre un archivo existente

Ejemplo:

```
rewrite(enteros);
reset(personas);
```

Operaciones

Cierre de archivos

```
close(nombre_logico);
```

Transfiere la información del buffer al disco.

Ejemplo:

```
close(enteros);
close(personas);
```

Operaciones

Lectura y escritura de archivos

```
read(nombre_logico, var_dato);
write(nombre_logico, var_dato);
```

El tipo de dato de la variable `var_dato` es igual al tipo de datos de los elementos del archivo.

Ejemplo:

```
read(enteros, e);
write(personas, p);
```

e : integer;

p : persona;

Operaciones adicionales

EOF(nombre_logico);
Controla el fin de archivo.

fileSize(nombre_logico);
Devuelve el tamaño de un archivo.

filePos(nombre_logico);
Devuelve la posición actual del puntero en el archivo.
En longitud fija, los registros se numeran de 0..N-1.

seek(nombre_logico, pos);
Establece la posición del puntero en el archivo.

Ejemplo

```
program creacion_archivo;
type
    persona = record
       dni: string[8]
       apellidoyNombre: string[30];
       direccion: string[40];
       sexo      : char;
       salario   : real;
    end;
    archivo_personas = file of persona;

var
    personas: archivo_personas;
    nombre_fisico: string[12];
    per: persona;
```

```
begin

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

  {enlace entre el nombre lógico y el nombre
   físico}
  assign(personas, nombre_fisico);

  {apertura del archivo para creación}
  rewrite(personas);
```

```
    {lectura del DNI una persona}
    write('Ingrese el dni de la persona: ');
    readln(per.dni);
    while (per.dni <> '') do begin
        {lectura del resto de los datos de la persona}
        write('Ingrese el apellido y nombre de la persona: ');
        readln(per.apellidoyNombre);
        write('Ingrese la dirección de la persona: ');
        readln(per.direccion);
        write('Ingrese el sexo de la persona: ');
        readln(per.sexo);
        write('Ingrese el salario de la persona: ');
        readln(per.salario);
        {escritura del registro de la persona en el archivo}
        write(personas, per);
        {lectura del DNI de una nueva persona}
        write('Ingrese otro dni o blanco para terminar: ');
        readln(per.dni);
    end;
    {cierre del archivo}
    close(personas);
end.
```

¿Dudas?

QR acceso provisorio al material de práctica