

Introducción a
Archivos

CONCEPTOS DE BASES DE DATOS

PRACTICA – INFORMACIÓN

Martes de 14:30 hs – 17:00 hs

Solo pueden rendir alumnos inscriptos en SIU

Fecha de examen prevista: 21/05

- Recuperatorio 1: 11/06
- Recuperatorio 2: 25/06

ARCHIVOS EN PASCAL

Tipos de archivos:

- ▶ **De longitud fija (File of <tipo>)**
- ▶ **De unidades indeterminadas (File)**
- ▶ **De Texto (Text):** caracteres estructurados en líneas ☐
lectura/escritura con conversión automática de tipos,
con acceso secuencial

ARCHIVOS EN PASCAL - OPERACIONES

- ▶ **Asignación (Assign):** una variable de tipo archivo representa a cualquier archivo del tipo definido \square se le debe asociar la ubicación física real del archivo para poder utilizar dicha variable.
- ▶ **Apertura:** establece la dirección de transferencia entre el programa y el archivo y la posición inicial del programa en el archivo
 - **Rewrite:** crea un archivo con el nombre asignado a la variable para escribir, y si ya existe borra su contenido, posicionando al programa al comienzo del archivo
 - **Reset:** abre el archivo con el nombre asignado a la variable tanto para leer como para escribir, posicionando al programa al comienzo del archivo
 - **Append:** abre el archivo de texto con el nombre asignado a la variable para escribir, posicionando al programa al final del archivo

ARCHIVOS EN PASCAL - OPERACIONES

- ▶ **Posicionamiento (Seek):** posiciona al programa en el número relativo de unidad que se especifica (no se permite en archivos de texto)
- ▶ **Lectura:** copia bytes desde archivo a la variable que se proporciona, en la posición donde está el programa dentro del archivo y posiciona al programa en la posición siguiente □ **Read** (archivos de registros y de texto), **ReadLn** (archivos de texto)
- ▶ **Escritura:** copia bytes desde la variable que se proporciona al archivo, en la posición donde está el programa en el archivo, y posiciona al programa en la siguiente posición □ **Write** (archivos de registros y de texto), **WriteLn** (archivos de texto)

ARCHIVOS EN PASCAL - OPERACIONES

- ▶ **Cierre (Close):** desactiva la dirección de transferencia entre el programa y el archivo y el posicionamiento del programa en el archivo.
- ▶ **Consultas:** permiten obtener características del archivo o de la posición del programa en el archivo
 - **FileSize** (tamaño en unidades del archivo)
 - **FilePos** (número relativo de unidad sobre la que está posicionado el programa en el archivo)
 - **EoF** (si el programa está posicionado a continuación de la última unidad del archivo)
 - **EoL** (si el programa está posicionado a continuación de la última línea del archivo -sólo archivos de texto)
- ▶ **Truncamiento (Truncate):** trunca el archivo a partir de la posición actual

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 mediante algún tipo de organización previa, independiente del orden físico.
- ▶ **Directo:** Un solo acceso para recuperar cada elemento.

EJEMPLO-ARCHIVO DE TEXTO- BINARIO

Type

tRegistroVotos=Record

codProv: integer;

codLoc: integer;

nroMesa: integer;

desc:string;

cantVotos: integer;

notas: string;

End;

tArchVotos=File of tRegistroVotos;

Var

opc: integer;

nomArch, nomArch2: String;

arch: tArchVotos;

carga: Text; // este archivo tiene los datos de los votos, se lee de el y genera el binario.

Además, luego se muestra como a partir del archivo de registros se escribe en el texto

votos: tRegistroVotos;

EJEMPLO-ARCHIVO DE TEXTO- BINARIO

begin

WriteLn('VOTOS');

WriteLn('0. Terminar el Programa');

WriteLn('1. Crear un archivo');

WriteLn('2. Abrir un archivo existente');

Repeat

Write('Ingresa el nro. de opcion: ');

ReadLn(opc);

If (opc=1) or (opc=2) then begin

WriteLn;

Write('Nombre del archivo de votos: ');

ReadLn(nomArch);

Assign(arch, nomArch);

Write('Nombre del archivo de texto: ');

ReadLn(nomArch2);

Assign(carga, nomArch2);

end;

EJEMPLO-ARCHIVO DE TEXTO- BINARIO

Case opc of

1: begin

Reset(carga);

Rewrite(arch);

while (not eof(carga)) do begin

 ReadLn(carga, votos.codProv, votos.codLoc, votos.nroMesa,
 votos.cantVotos, desc);

 ReadLn(carga, votos.notas);

 Write(arch, votos);

end;

Write('Archivo cargado. Oprima tecla de ingreso para continuar...');

ReadLn;

Close(arch); Close(carga)

end;

10

Qué sucede cuando hay más de un campo string en el archivo de texto?

EJEMPLO-ARCHIVO DE BINARIO - TEXTO

2: begin

Reset(arch);

Rewrite(carga);

WriteLn;

While (not eof(arch)) do begin

Read(arch, votos);

WriteLn(carga, votos.codProv, votos.codLoc, votos.nroMesa, votos.cantVotos,
votos.desc);

WriteLn(carga, votos.notas);

end;

Close(arch); Close(carga);

WriteLn;

Write('Oprima cualquier tecla para continuar...');

ReadLn

end;

until (opc=0);

End.