Facultad de Ciencias UNER de la Administración

Algoritmos y Programación

Estructuras de Datos - Archivos -

Prof. Miguel A. Fernández

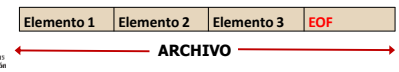
Archivos

Un archivo o fichero (file) es una estructura de datos que reside en memoria secundaria.

Consiste en un conjunto de informaciones estructuradas en unidades de acceso denominadas registros, todos del mismo tipo y en número indeterminado.

Permiten superar las limitaciones impuestas por la memoria principal.

En Pascal, un archivo es una **secuencia de elementos** que pertenecen al mismo tipo o estructura. Es decir, puede ser una secuencia de caracteres, números o registros, por lo que su representación lógica puede hacerse como una secuencia de módulos de igual tamaño.

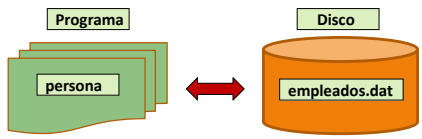


Archivos

Cada archivo es referenciado por un identificador.

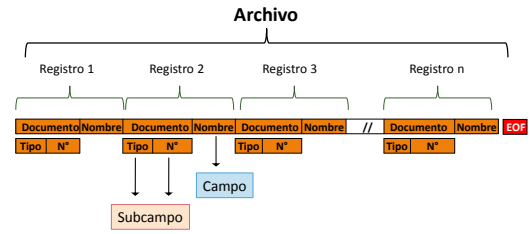
El identificador del archivo en un programa puede ser diferente al que figura en el medio de almacenamiento.

Normalmente al identificador del medio de almacenamiento se le adiciona una extensión.



Archivos

Esquema General de un Archivo

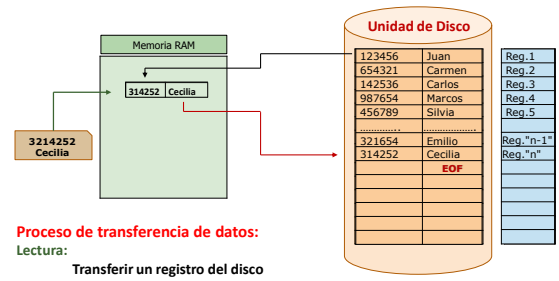


Archivos

CARACTERISTICAS DE LOS ARCHIVOS

- **Residen en soportes de información externos.**
- **Son independientes respecto de los programas.**
- **Pueden ser accedidos por diferentes programas.**
- **Permanencia de las informaciones almacenadas.**
- **Gran capacidad de almacenamiento.**

Archivos



Proceso de transferencia de datos:

Lectura:
Transferir un registro del disco a la memoria RAM.

Grabación:
Transferir un registro de la memoria RAM al disco.

Archivos

LOS REGISTROS:

Similar a una ficha de papel.

Contiene información relativa a una entidad (persona, objeto, hecho, etc.

Cada registro tiene la misma estructura.

Los datos individuales dentro del registro se denominan "campos".

Los campos pueden ser de longitud fija o variable.

Facultad de Ciencias UNER de la Administración

Archivos

Tipos de registros.

Registro lógico:

Cada uno de los componentes del archivo, conteniendo el conjunto de informaciones que se tratan de manera unitaria.

Está constituido por uno o más elementos denominados campos, que pueden ser de diferentes tipos y que a su vez pueden estar compuestos por subcampos.

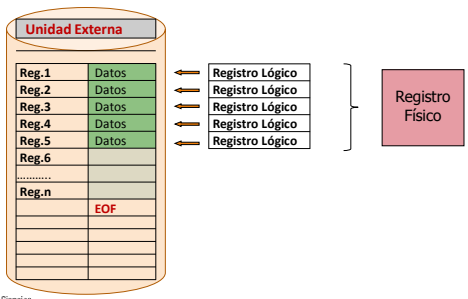
Registro físico o *bloque:*

Corresponde a la cantidad de información que se transfiere en cada operación de acceso (lectura o escritura).

Facultad de Ciencias UNER de la Administración

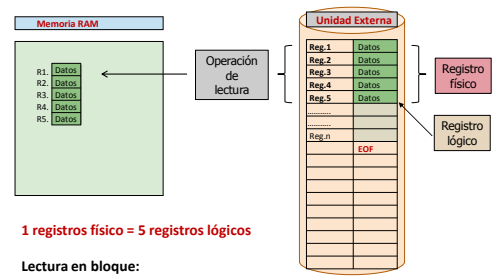
Archivos

1 registro físico = 5 registros lógicos



Facultad de Ciencias UNER de la Administración

Archivos

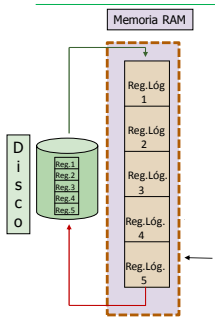


1 registros físico = 5 registros lógicos

Lectura en bloque:
Se leen 5 registros lógicos juntos y se los "copian" en memoria.

Facultad de Ciencias UNER de la Administración

Archivos



EN UN PROCESO DE ACTUALIZACIÓN

1. **Lectura**: Se lee el registro físico del disco. Se graba en memoria y se apunta al primer registro lógico.
   ReGrabación: No se transfiere a disco
2. **Lectura**: Se apunta al segundo registro lógico
   ReGrabación: No se transfiere a disco
3. **Lectura**: Se apunta al tercer registro lógico
   ReGrabación: No se transfiere a disco
4. **Lectura**: Se apunta al cuarto registro lógico
   ReGrabación: No se transfiere a disco
5. **Lectura**: Se apunta al quinto registro lógico
   ReGrabación: Se transfiere el registro físico a disco

Facultad de Ciencias UNER de la Administración

Archivos

CLASIFICACIÓN DE LOS ARCHIVOS

En función del contenido:

a) Archivo de **datos**
b) Archivo de **programa**
c) Archivo de **texto**

En función del tipos de acceso:

a) Acceso secuencial
Los registros se graban uno a continuación del otro. De la misma manera se leen. Es decir que para leer el registro 'n' es necesario leer los 'n - 1' registros anteriores.

b) Acceso directo
Permiten el acceso directo a un registro.

Facultad de Ciencias UNER de la Administración

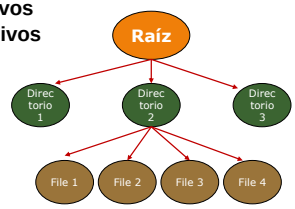
Archivos

EL SISTEMA DE GESTIÓN DE ARCHIVOS

Metodología utilizada por el sistema operativo para el almacenamiento y organización de archivos en una computadora.

- **FAT** (File allocate Table)
- **NTFS** (New Technology File System)

- La creación de archivos
- El directorio de archivos



Facultad de Ciencias UNER de la Administración

Archivos

Sintaxis de las operaciones fundamentales:

1. **Apertura y cierre:**
   ABRIR <nombre de archivo>
   CERRAR <nombre de archivo>
2. **Lectura de un registro**:
   LEER <nombre de Archivo>
3. **Grabación de un registro**:
   GRABAR <nombre de Archivo, nombre registro>
4. **Re-Grabación de un registro**:
   REGRABAR <nombre de Archivo, nombre registro>
5. **Eliminar un Registro:**
   Borrar <nombre de Archivo, nombre registro>
6. **EOF(nombre de archivo):**
   Retorna TRUE si se llegó al final de un archivo en un intento de lectura y FALSE en caso contrario.

Archivos

Como especificamos archivos:

Se desea crear un archivo el que se guarden el Nro.de Cuenta, Apellido y nombres, y saldo que adeuda de un grupo de clientes de una empresa.

El primer paso a realizar es crear un registro que contenga todos estos campos:

En pseudocódigo lo especificamos:

```
Nombre: <clientes>
Acceso: secuencial
Registro:
        CUENTA - Nro. de Cuenta
        NOMBRE- Apellido y Nombres
        SALDO  - Saldo
```

Facultad de Ciencias UNER de la Administración

Archivos

Como definir archivos en Pascal:

1. Se define el registro y el archivo de tipo registro:

```
TYPE
    (nombre-tipo-Registro) = RECORD
        Cuenta : Integer;
        Nombre : String[40];
        Saldo : Real
    END;
    (nombre-tipo-Archivo) = file of (nombre-tipo-Registro);
```

2. El siguiente paso es declarar las variables del tipo archivo y tipo registro

```
VAR
    (nombre-variable-archivo) : (Nombre-tipo-Archivo);
    (nombre-variable-registro) : (Nombre-tipo-Registro);
```

3. Se le asigna un nombre real para guardarlo en el disco, mediante una primitiva.

```
Assign (Nombre-variable-Archivo, 'Nombre.ext');
```

UNER

Archivos

Caso 1

Generar un archivo con los datos de los clientes de una empresa, siguiendo el ejemplo anterior.

```
Program Clientes
Type
    Tregistro = record
        cuenta : Integer;
        nombre: string[40];
        saldo : real
        end;
    Tarchivo = file of Tregistro;
Var
    Clientes : Tarchivo;
    Reg : Tregistro;
```

Facultad de Ciencias UNER de la Administración

Archivos

Caso 1

Generar un archivo con los datos de los clientes de una empresa, siguiendo el ejemplo anterior.

```
Program Clientes
Type
    Tregistro = record
        cuenta : Integer;
        nombre: string[40];
        saldo : real
        end;
    Tarchivo = file of Tregistro;
Var
    Clientes : Tarchivo;
    Reg : Tregistro;
```

```
Begin
    Assign (Clientes, "Cliente.dat")
    Abrir (Clientes)
    Ingresar Reg.cuenta
    Mientras Reg.cuenta <> 0
        Ingresar Reg.nombre, Reg.saldo
        Grabar (Clientes, Reg)
        Ingresar Reg.cuenta
    FinMientras
    Cerrar (Clientes)
End.
```

UNER

Archivos

Caso 2

Se desea imprimir el contenido del archivo generado en el ejemplo anterior:

```
INICIO
Abrir (Clientes)
Leer (Clientes)
Mientras NOFIN (Clientes)
        Imprimir: Reg.cuenta,
                  Reg.nombre,
                  Reg.saldo
        Leer (Clientes)
Finmientras
Cerrar (Clientes)
FIN
```

Facultad de Ciencias UNER de la Administración

Archivos

Caso 3

Se desea imprimir los datos de un empleado del archivo generado en el ejemplo anterior, para lo cual el operador ingresa el N° de Cuenta.

```
INICIO
Ingresar TNC
Mientras TNC <> 0
    Abrir (Clientes)
    ESTA := False
    Leer (Clientes)
    Mientras NOEOF <Clientes> y ESTA = False
        Si TNC = Reg.cuenta
            Mostrar reg.nombre, reg.saldo
            ESTA := True
        Sino
            Leer (Clientes)
        FinSi
    FinMientras
    Si ESTA = False
        Mostrar "No existe Cuenta"
    FinSi
    Cerrar (Clientes)
    Ingresar TNC
FinMientras
FIN
```

Facultad de Ciencias UNER de la Administración

Archivos

Caso 4

Se desea actualizar los saldos de los clientes.

Por cada operación ingresan por teclado:

- TNC - Nro.de Cuenta
- TCOD - Código de Operación [1=Compra, 2=Pago]
- TIMP - Importe

```
INICIO
Ingresar TNC
Mientras TNC <> 0
    Abrir (Clientes)
    ESTA := False
    Leer (Clientes)
    Mientras NOEOF (Clientes) y ESTA = False
        Si TNC = reg.cuenta
            Ingresar TCOD, TIMP
            Si TCOD = 1
                Reg.saldo := Reg.saldo + TIMP
            Sino
                Reg.saldo := Reg.saldo – TIMP
            FinSi
            Regrabar (Clientes, Reg)
            ESTA := True
        Sino
            Leer (Clientes)
        FinSi
    FinMientras
    Si ESTA = False
        Mostrar "No existe Cuenta"
    FinSi
    Cerrar (Clientes)
    Ingresar TNC
FinMientras
FIN
```

Facultad de Ciencias UNER de la Administración

Archivos

Caso 5

Se desea Imprimir un listado ordenado por Apellido y Nombre de los clientes.

Procedimiento:

1) Pasar los datos del archivo a un array
2) Ordenar
3) Imprimir

```
Program Clientes
Type
  Tregistro = record
     cuenta : Integer;
     nom: string[40];
     saldo : real
     end;
  Tarchivo = file of Tregistro;
  Tvector = array [1..500] of Tregistro;
Var
  Clientes : Tarchivo;
  TReg : Tregistro;
  VEC : Tvector;
```

Archivos (Solución)

```
Procedimiento Imprime
Var
   DIM, TAM, j, i : Integer;
   AUX :  Tregistro;
Begin

     Mostrar «Procesando»
     Abrir (Clientes)
     Leer (Clientes)
     i := 0
     Mientras NoEOF (Clientes)
        i := i + 1
        VEC(i) := Treg
        Leer (Clientes)
     FinMientras
     Cerrar (Clientes)
     TAM := i
```

```
DIM := TAM / 2
Mientras DIM <> 0
    i := 0
    Mientras i < (TAM – DIM)
        i := i + 1
        j := i
        Mientras i > 0 and then VEC(i).nom > VEC(i+DIM).nom
            AUX := VEC(i)
            VEC(i) := VEC(i+DIM)
            VEC(i+DIM) := AUX
            i := i -DIM
        FinMientras
        i := j
    FinMientras
    DIM := DIM / 2
FinMientras

Para i = 1, 75, 1
    Imprimir VEC(I).cuenta, VEC(i).nom, VEC(i).saldo
FinPara

End-Procedimiento Imprime.
```

Facultad de Ciencias UNER de la Administración

Archivos - Ejercicio

Codificar un programa para grabar y consultar los datos del personal administrativo de una empresa, contemplando los siguientes módulos:

Módulo 1.

Ingresar los siguientes datos de cada empleado:

NUL - Número de legajo
APE - Apellido y Nombre
NUD - Nro.de documento
CAT - Categoría (rango de 1 a 17)

Estos datos grabarlos en un archivo secuencial (EMPLEADOS)

Módulo 2.

Consultar los datos del archivo. El operador digitará el número de legajo del empleado y se debe mostrar todos los datos del empleado.

Luego de mostrar los datos de un empleado consultado, se debe mostrar la cantidad de empleados que hay en la empresa con la misma categoría que el empleado consultado.

Facultad de Ciencias UNER de la Administración

Archivos (Solución Ejercicio)

```
Program Clientes
Type
   Aregistro = record
        NUL : Integer;
        APE : string[40];
        NUD : Integer;
        CAT  : Integer
        end;
   Aarchivo = file of Aregistro;
   Vvector = array [1..17] of integer;
Var
   Empleado : Aarchivo;
   Preg : Aregistro;
   Vcat : Vvector;
    i, OPC, NULE : integer;
    LOK : Boolean;
```

```
Begin
    Assign (Empleado, "Emple.dat")
    Para i = 1,17,1
        Vcat(i) := 0
    FinPara
    MENU
    Mientras OPC <> 0
        Según OPC
            1 : Ingreso
            2 : Consulta
        FinSegún
        MENU
     FinMientras
End.
```

```
Procedimiento Menu
   Mostar "Seleccione una Opción"
   Mostrar "0 – Fin de Tarea"
           "1 – Ingresa Datos"
           "2 – Consulta Datos"
   Ingresar OPC
End-Procedimiento Menu.
```

Facultad de Ciencias UNER de la Administración

Archivos (Solución Ejercicio)

```
Procedimiento Ingreso
  Abrir (Empleado)
  Ingresar Preg.NUL
  Mientras Preg.NUL <> 0
       Ingresar Preg.APE, Preg.NUD,
                Preg.CAT
       Grabar (Empleado, Preg)
       Vcat(Preg.CAT) := Vcat(Preg.CAT) + 1
       Ingresar Preg.NUL
  FinMientras
  Cerrar (Empleado)
End-Procedimiento Ingreso
```

```
Procedimiento Consulta
   Ingresar NULE
   Mientras NULE <> 0
        Abrir (Empleado)
        LOK := False
        Leer (Empleado)
        Mientras NoEOF (Empleado) and LOK = False
             Si NULE = Preg.NUL
                  Mostrar Preg.APE, Preg.NUD
                          Preg.CAT
                  Mostrar "Cant.Empleados = Categoría"
                  Mostrar Vcat(Preg.CAT)
                  LOK := True
             Sino
                  Leer (Empleado)
             FinSi
        FinMientras
        Si LOK = False
             Mostrar "Dato no Encontrado"
        FinSi
        Cerrar (Empleado)
        Ingresar NULE
   FinMientras
End-Procedimiento Consulta.
```

Facultad de Ciencias UNER de la Administración

Archivos - Ejercicio

ALGUNAS CUESTIONES DE DISEÑO:

Cohesión y acoplamiento de módulos

¿ Qué ocurre si, luego de ingresar los datos, se termina el programa y posteriormente se vuelve a ingresar para consultar datos solamente ?

¡ En este caso no se tienen los datos guardados en el array !

¿ Cómo solucionamos el problema ?

Una alternativa:
Acumular en el módulo de consulta

Facultad de Ciencias UNER de la Administración

Archivos (Solución Ejercicio)

```
Program Clientes
Type
   Aregistro = record
        NUL : Integer;
        APE : string[40];
        NUD : Integer;
        CAT  : Integer
        end;
   Aarchivo = file of Aregistro;
   Vvector = array [1..17] of integer;
Var
   Empleado : Aarchivo;
   Preg : Aregistro;
   Vcat : Vvector;
    i, OPC, NULE : integer;
    LOK : Boolean;
```

```
Begin
    Assign (Empleado, "Emple.dat")
    Para i = 1,17,1
        Vcat(i) := 0
    FinPara
    MENU
    Mientras OPC <> 0
        Según OPC
            1 : Ingreso
            2 : Consulta
        FinSegún
        MENU
     FinMientras
End.
```

```
Procedimiento Menu
   Mostar "Seleccione una Opción"
   Mostrar "0 – Fin de Tarea"
           "1 – Ingresa Datos"
           "2 – Consulta Datos"
   Ingresar OPC
End-Procedimiento Menu.
```

Facultad de Ciencias UNER de la Administración

Archivos (Solución Ejercicio)

```
Procedimiento Ingreso
  Abrir (Empleado)
  Ingresar Preg.NUL
  Mientras Preg.NUL <> 0
       Ingresar Preg.APE, Preg.NUD,
                Preg.CAT
       Grabar (Empleado, Preg)
       Ingresar Preg.NUL
  FinMientras
  Cerrar (Empleado)
End-Procedimiento Ingreso
```

```
Procedimiento Consulta
   Mostrar «Aguarde un momento»
   Abrir (Empleado)
   Leer (Empleado)
   Mientras NoEOF (Empleado)
        Vcat(Preg.CAT) := Vcat(Preg.CAT) + 1
        Leer (Empleado)
   FinMientras
   Cerrar (Empleado)
```

```
   Ingresar NULE
   Mientras NULE <> 0
        Abrir (Empleado)
        LOK := False
        Leer (Empleado)
        Mientras NoEOF (Empleado) and LOK = False
             Si NULE = Preg.NUL
                  Mostrar Preg.APE, Preg.NUD
                          Preg.CAT
                  Mostrar "Cant.Empleados = Categoría"
                  Mostrar Vcat(Preg.CAT)
                  LOK := True
             Sino
                  Leer (Empleado)
             FinSi
        FinMientras
        Si LOK = False
             Mostrar "Dato no Encontrado"
        FinSi
        Cerrar (Empleado)
        Ingresar NULE
   FinMientras
End-Procedimiento Consulta.
```

Facultad de Ciencias UNER de la Administración

Archivos - Ejercicio

El tema de la cohesión y acoplamiento de módulos:

¿ Qué ocurre si, luego de ingresar a una consulta se sale del módulo y en el menú se vuelve a ingresar al mismo módulo ?

¡ Se vuelve a recorrer el archivo para almacenar los datos al array !

¿ Cómo solucionamos el problema ?

Una alternativa:
- Consultar si el array tiene datos

Facultad de Ciencias UNER de la Administración

Archivos (Solución Ejercicio)

```
Program Clientes
Type
   Aregistro = record
        NUL : Integer;
        APE : string[40];
        NUD : Integer;
        CAT  : Integer
        end;
   Aarchivo = file of Aregistro;
   Vvector = array [1..17] of integer;
Var
   Empleado : Aarchivo;
   Preg : Aregistro;
   Vcat : Vvector
    i, OPC, NULE : integer;
    LOK : Boolean;
    VB : Boolean;
```

```
Begin
   Assign (Empleado, "Emple.dat")
   Para i = 1,17,1
      Vcat(i) := 0
   FinPara
   VB := False
   MENU
   Mientras OPC <> 0
        Según OPC
            1 : Ingreso
            2 : Consulta
        FinSegún
        MENU
    FinMientras
End.
```

```
Procedimiento Menu
   Mostar "Seleccione una Opción"
   Mostrar "0 – Fin de Tarea"
           "1 – Ingresa Datos"
           "2 – Consulta Datos"
   Ingresar OPC
End-Procedimiento Menu.
```

Facultad de Ciencias UNER de la Administración

Archivos (Solución Ejercicio)

```
Procedimiento Ingreso
  Abrir (Empleado)
  Ingresar Preg.NUL
  Mientras Preg.NUL <> 0
       Ingresar Preg.APE, Preg.NUD,
                Preg.CAT
       Grabar (Empleado, Preg)
       Ingresar Preg.NUL
  FinMientras
  Cerrar (Empleado)
End-Procedimiento Ingreso
```

```
Procedimiento Consulta
  Si VB = False
     Mostrar «Aguarde un momento»
     Abrir (Empleado)
     Leer (Empleado)
     Mientras NoEOF (Empleado)
        Vcat(Preg.CAT) := Vcat(Preg.CAT) + 1
        Leer (Empleado)
     FinMientras
     Cerrar (Empleado)
     VB := True
  FinSi
```

```
   Ingresar NULE
   Mientras NULE <> 0
      Abrir (Empleado)
      LOK := False
      Leer (Empleado)
      Mientras NoEOF (Empleado) and LOK = False
         Si NULE = Preg.NUL
            Mostrar Preg.APE, Preg.NUD
                    Preg.CAT
            Mostrar "Cant.Empleados = Categoría"
            Mostrar Vcat(Preg.CAT)
            LOK := True
         Sino
            Leer (Empleado)
         FinSi
      FinMientras
      Si LOK = False
         Mostrar "Dato no Encontrado"
      FinSi
      Cerrar (Empleado)
      Ingresar NULE
   FinMientras
End-Procedimiento Consulta.
```

Archivos - Ejercicio

El tema de la cohesión y acoplamiento de módulos:

¿ Qué ocurre si se sigue la siguiente secuencia de acciones, sin salir del programa ?

1. Se ingresan los datos de empleados
2. Se selecciona la opción consulta
3. Se vuelve a ingresar datos de mas empleados
4. Se vuelve a seleccionar la opción de consulta

¡ El array no va a contener los datos de los empleados que ingresaron según punto 3 !

¿ Cómo solucionamos el problema ?

Facultad de Ciencias UNER de la Administración

Archivos (Solución)

```
Begin
   Assign (Empleado, "Emple.dat")
   VB := False
   MENU
   Mientras OPC <> 0
        Según OPC
            1 : Ingreso
            2 : Consulta
        FinSegún
        MENU
    FinMientras
End.
```

No se inicializa el array

```
Procedimiento Menu
   Mostar "Seleccione una Opción"
   Mostrar "0 – Fin de Tarea"
           "1 – Ingresa Datos"
           "2 – Consulta Datos"
   Ingresar OPC
End-Procedimiento Menu.
```

```
Procedimiento Ingreso
  Abrir (Empleado)
  Ingresar Preg.NUL
  Mientras Preg.NUL <> 0
       Ingresar Preg.APE, Preg.NUD,
                Preg.CAT
       Grabar (Empleado, Preg)
       VB := False
       Ingresar Preg.NUL
  FinMientras
  Cerrar (Empleado)
End-Procedimiento Ingreso
```

Facultad de Ciencias UNER de la Administración

Archivos (Solución)

Verificar con distintas secuencias de acciones.

```
Procedimiento Consulta
  Si VB = False
     Para i = 1,17,1
        Vcat(i) := 0
     FinPara
     Mostrar «Aguarde un momento»
     Abrir (Empleado)
     Leer (Empleado)
     Mientras NoEOF (Empleado)
        Vcat(Preg.CAT) := Vcat(Preg.CAT) + 1
        Leer (Empleado)
     FinMientras
     Cerrar (Empleado)
     VB := True
  FinSi
```

```
   Ingresar NULE
   Mientras NULE <> 0
      Abrir (Empleado)
      LOK := False
      Leer (Empleado)
      Mientras NoEOF (Empleado) and LOK = False
         Si NULE = Preg.NUL
            Mostrar Preg.APE, Preg.NUD
                    Preg.CAT
            Mostrar "Cant.Empleados = Categoría"
            Mostrar Vcat(Preg.CAT)
            LOK := True
         Sino
            Leer (Empleado)
         FinSi
      FinMientras
      Si LOK = False
         Mostrar "Dato no Encontrado"
      FinSi
      Cerrar (Empleado)
      Ingresar NULE
   FinMientras
End-Procedimiento Consulta.
```

Facultad de Ciencias UNER de la Administración

Archivos - Ejercicio

Se deben procesar los datos del uso de los 75 teléfonos internos fijos disponibles en una empresa. Se requiere codificar los siguientes módulos:

1. Ingreso de datos: De cada línea se ingresa por teclado:
- NUL – Número de Línea
- OFI – Oficina donde está instalada la línea.
- RES – Apellido y nombre del responsable
- MIS – Minutos acumulados por llamadas salientes
- MIE – Minutos acumulados por llamadas entrantes.

El ingreso se realiza ordenado por número de línea.
Con estos datos grabar un archivo <TELEFONO> de acceso secuencial.

2. Proceso llamadas: De cada llamada a/de la línea se ingresa:
- NUM – Número de línea
- COL – Código de llamada (1-Entrante, 2-Saliente)
- CAM – Cantidad de minutos que dura la llamada

Con estos datos actualizar los campos que correspondan del archivo

3. Listado: Imprimir un listado ordenado por Oficina donde se encuentra la línea que contenga:

Oficina – Número de Línea – Apellido y Nombre responsable

Archivos secuenciales

FIN

UNER