

Almacenamiento Externo

- Operaciones adicionales sobre ficheros binarios
 - Para posicionar el marcador en cualquier registro
 - APUNTAR(varfich, posición)
 - La posición es un entero o una variable de ese tipo
 - El rango es [0, numReg-1]
 - Para conocer en que posición se encuentra nuestro marcador .
 - Posición(varfich) //entero



Almacenamiento Externo

- Operaciones adicionales sobre ficheros binarios
 - Para conocer el número total de registros de un fichero
 - Tamaño(varfich)//
 - Ejemplo:

TAMAÑO (vfic): entero

```
ABRIR(F1)
P ← POSICION(F1)
T ← TAMAÑO(F1)
SI T > 1 ENTONCES
    APUNTAR(F1, T-1)
    ESCRIBIR(F1, fecha)
    APUNTAR(F1, P)
FIN_SI
```

```
ABRIR(F1)
APUNTAR(F1, TAMAÑO(F1))
ESCRIBIR(F1, fecha)
CERRAR(F1)
```



Almacenamiento Externo

- Resumen Operaciones sobre ficheros binarios

■ Declaración	Vfic:FICHERO DE tipo_datos
■ Asignación	ASIGNAR(vfic,cadena)
■ Creación	CREAR(vfic)
■ Apertura	ABRIR(vfic)
■ Lectura de datos	LEER(vfic,var)
■ Escritura de datos	ESCRIBIR(vfic,var)
■ Cierre	CERRAR(vfic)
■ Apuntar	APUNTAR(vfic,var_entera)
■ Posición del apuntador	POSICION(vfic)
■ Tamaño del fichero	TAMAÑO(vfic)



Almacenamiento Externo

- Ejercicio
 - Desarrolle un algoritmo que permita almacenar 50 alumnos en un fichero binario.



Almacenamiento Externo

- Añadir Registros

```
VARIABLES
  emp:TEmpleado
  F_emp:TF_emp
  resp:CARACTER
INICIO
  ASIGNAR(F_emp,nombref)
  ABRIR(F_emp)
  SI IOResult <> 0 ENTONCES
    ESCRIBIR("ERROR AL ABRIR FICHERO")
  SINO
    APUNTAR(F_emp,TAMAÑO(F_emp))
    REPETIR
      lee_reg(emp)  {--- procedimiento para leer desde teclado los datos ---}
      ESCRIBIR(F_emp,emp)
      ESCRIBIR("Registro insertado. Desea continuar (S/N)? ")
      resp ← leetecla
    HASTA UPGASE(resp)='N'
  FIN_SI
  CERRAR(F_emp)
FIN_PROC
```



Almacenamiento Externo

- Modificar un registro

```
VARIABLES
  emp_mod,emp:TEmpleado      F_emp:TF_emp
  cod:ENTERO                 encontrado:LOGICO
INICIO
  ASIGNAR(F_emp,nombref)
  ABRIR(F_emp)
  SI IOResult <> 0 ENTONCES ESCRIBIR('ERROR AL ABRIR FICHERO')
  SINO
    ESCRIBIR('Introduzca la clave del empleado a modificar:')
    LEER(cod)
    encontrado ← false
    MIENTRAS NOT EOF(f_emp) AND NOT encontrado HACER
      LEER(f_emp,emp)
      SI (emp.codigo=cod) ENTONCES
        encontrado ← true
        APUNTAR(F_emp,POSICION(F_emp)-1)
        lee_reg(emp_mod)      {-- módulo que lee los valores no clave de un registro --}
        emp_mod.codigo ← cod
        ESCRIBIR(f_emp,emp_mod)
      FIN_SI
    FIN_MIENTRAS
    SI NOT encontrado ENTONCES
      ESCRIBIR('Registro no encontrado')
    FIN_SI
  CERRAR(F_emp)
```



Almacenamiento Externo

- Consultar un registro

```
VARIABLES
  emp:Empleado          F_emp:TF_emp
  cod:ENTERO             encontrado:LOGICO
INICIO
  ASIGNAR(F_emp,nombref)
  ABRIR(F_emp)
  SI IOResult <> 0 ENTONCES ESCRIBIR('ERROR AL ABRIR FICHERO')
  SINO
    ESCRIBIR('Introduzca la clave del empleado a consultar:')
    LEER(cod)
    encontrado ← false
    MIENTRAS NOT EOF(f_emp) AND NOT encontrado HACER
      LEER(f_emp,emp)
      SI (emp.codigo=cod) ENTONCES
        encontrado ← true
        muestra_reg(emp)  {--- Procedimiento que muestra los datos de un registro ---}
      FIN_SI
    FIN_MIENTRAS
    SI NOT encontrado ENTONCES
      ESCRIBIR('Registro no encontrado ')
    FIN_SI
  CERRAR(F_emp)
FIN_SI
```



Almacenamiento Externo

- Resumen

- Para binarios y texto

- Asignar: Asignar(varfich,cadena)
 - Apertura: Abrir(varfich)
 - Creación: Crear(varfich)
 - Escritura: Escribir(varfich)
 - Cierre: Cerrar(varfich)
 - Fin de archivo: EOF(varfich)



Almacenamiento Externo

- Sólo para ficheros de texto
 - Declaración:
 - Variables varfich: TEXT
 - Para añadir al final del archivo que se encuentra creado.
 - Añadir(varfich)
 - Lectura y escritura con salto de línea
 - LeerLN(varfich, lista parámetros)
 - EscribirLN(varfich, lista parámetros)



Almacenamiento Externo

Realizar un pseudocódigo que permita almacenar 100 registros y consultar por un alumno en particular

