

# Programación II

Mantenimiento de Archivos

# Mantenimiento de archivos

## ABML

Todos los sistemas de procesamiento de transacciones que operan con un conjunto de archivos relacionados —como los vistos en la clase anterior— realizan tareas de mantenimiento sobre dichos archivos.

Estas tareas se conocen como ABM, siglas de **A**lta, **B**aja y **M**odificación. Además de estas funciones básicas, resulta muy útil incorporar la posibilidad de generar **L**istados.

# Mantenimiento de archivos: Altas

La operación de **alta** consiste en agregar nuevos registros a un archivo. En su forma más básica, este proceso incluye los siguientes pasos:

1. **Apertura del archivo** en modo escritura ("wb") o anexo ("ab"), según corresponda.
2. **Asignación de datos** al registro, cargando la información en la variable de memoria correspondiente.
3. **Escritura del registro** en el archivo.
4. **Cierre del archivo** una vez finalizada la operación.

# Mantenimiento de archivos: Bajas

La operación de **baja** consiste en eliminar registros de un archivo. Esta es una de las tareas más delicadas dentro del mantenimiento, ya que la eliminación de información puede ser irreversible y ocasionar serios problemas si no se realiza correctamente.

Existen dos tipos de bajas:

1. Baja lógica.
2. Baja física.

# Mantenimiento de archivos: Bajas

## 1. BAJA LÓGICA

- En una **baja lógica**, el registro no se elimina físicamente del archivo. En lugar de ello, se “**marca**” como inactivo o eliminado mediante un indicador, como un campo de **estado**.
- Al agregar un nuevo registro, el campo de estado se establece en **true**.
- Al realizar la baja lógica, el valor de este campo cambia a **false**.
- Este método permite que un registro marcado como eliminado se pueda recuperar, ya que no ha sido borrado físicamente del archivo. En la práctica, la baja lógica consiste en modificar el valor de este campo de estado.
- Esta técnica permite mantener un historial completo y facilita la recuperación del registro en caso de que sea necesario.

# Mantenimiento de archivos: Bajas

## 2. BAJA FÍSICA

La **baja física** consiste en eliminar completamente un registro del archivo. Este proceso no permite eliminar registros directamente, por lo que es necesario generar un nuevo archivo con los registros válidos, omitiendo aquellos marcados para eliminación.

El procedimiento habitual es el siguiente:

1. Se realiza una copia del archivo, registro por registro, con una extensión diferente (por ejemplo, .bak).
2. Se abre el archivo original en modo escritura ("wb") para eliminar los registros existentes.
3. Finalmente, se copian al archivo original los registros que se desean conservar desde el archivo .bak.

# Mantenimiento de archivos: Modificación

La **modificación** consiste en cambiar el valor de uno o más campos de un registro en el archivo. Este proceso se puede describir con los siguientes pasos:

1. Localizar la posición del registro que se desea modificar.
2. Copiar el registro a una variable de memoria.
3. Asignar los nuevos valores a los campos correspondientes.
4. Escribir el registro modificado de nuevo en la misma posición.

# Mantenimiento de archivos: Modificación

Es importante tener en cuenta que, para localizar un registro, el programa debe haber leído previamente los registros anteriores. Esto se debe a que, al encontrar el registro deseado, el puntero de archivo **FILE** se posiciona al inicio del siguiente registro. Para modificar un registro, es necesario ubicar el puntero **FILE** en la posición correcta. Esto se logra utilizando la función *fseek()*, cuya sintaxis es:

*fseek(punteroFile, desplazamiento, desde\_donde\_desplazar)*

1. **punteroFile**: El puntero al archivo.
2. **desplazamiento**: El número de bytes a mover el puntero.
3. **desde\_donde\_desplazar**: La posición desde la que se hará el desplazamiento (usualmente SEEK\_SET, SEEK\_CUR, o SEEK\_END).



# Mantenimiento de archivos: Modificación

Una vez localizado el registro, es necesario retroceder una posición para acceder al registro anterior. Una manera de hacerlo es mediante la instrucción:

```
fseek(pFILE, -sizeof(registro), SEEK_CUR);
```

Esta instrucción indica que se debe mover el puntero de archivo hacia atrás la cantidad de bytes que ocupa un registro, tomando como referencia la posición actual.

Si bien esta forma es válida, debe tenerse en cuenta que *algunos compiladores no permiten utilizar valores negativos en el desplazamiento del fseek()*, lo cual puede generar incompatibilidades.

# Mantenimiento de archivos: Modificación

Para resolver este inconveniente se utiliza la función *ftell()*, cuyo prototipo es:

```
long ftell(FILE *p);
```

Esta función devuelve la cantidad de bytes existentes entre el inicio del archivo y la posición actual del puntero. De esta manera, la llamada anterior a *fseek()* podría reescribirse de la siguiente forma:

```
fseek(pFILE, ftell(pFILE) - tamaño_registro, SEEK_SET);
```

# Mantenimiento de archivos: Listados

Los listados se utilizan para mostrar la información almacenada en un archivo.

- **Listado básico:** consiste en leer secuencialmente cada uno de los registros de un archivo e imprimirlos en pantalla.
- **Listado filtrado:** permite mostrar únicamente aquellos registros que cumplen con una determinada condición, por ejemplo: productos con precio superior a un valor dado, artículos con un stock menor a cierta cantidad, entre otros.