

Ficheros. Tipos.
Características.
Organizaciones

TEMA 13

ABACUS NT

Oposiciones 2021

Índice

- 1. Introducción**
- 2. Conceptos y características**
- 3. Operaciones con ficheros**
- 4. Tipos de ficheros**
 - 4.1. Segundo su función.**
 - 4.1.1. Permanentes.
 - 4.1.2. Temporales.
 - 4.2. Segundo su contenido.**
 - 4.2.1. Ficheros de texto
 - 4.2.2. Ficheros binarios
 - 4.3. Segundo la longitud de los registros.**
 - 4.4. Segundo el tipo de acceso**
- 5. Organización de ficheros**
 - 5.1. Organización secuencial**
 - 5.2. Organización secuencial encadenada**
 - 5.3. Organización secuencial indexada**
 - 5.4. Organización directa o aleatoria**
 - 5.5. Ficheros con organización secuencial indexada encadenada**
- 6. Conclusión**
 - 6.1. Relación del tema con el sistema educativo actual**
- 7. Bibliografía**

1. Introducción

La información en la sociedad actual es un aspecto relevante en múltiples ramas del conocimiento; incluyendo la biología genética, la física cuántica, la divulgación científica y por supuesto la informática, Internet y el tratamiento de la información.

Cada día, se intercambian unos 2,5 billones de bytes de datos. Se estima que el 90% de los datos existentes en el mundo actual se han generado en el curso de los últimos dos años. Este diluvio se conoce como **Big Data**.

Nos enfrentamos al reto de almacenar de forma fiable, duradera y fácilmente transmisible, **exabytes** de datos. Para ello necesitamos soportes que mantengan la información más allá del ciclo de trabajo de un ordenador, y una estructura de datos que nos garantice su acceso.

Los ficheros son la estructura de datos en los que se apoya el almacenamiento en memoria secundaria. Tanto programas como datos, incluyendo bases de datos completas y copias de seguridad, se almacenan en última instancia en soportes auxiliares y por tanto en ficheros. El tipo de ficheros que utilicemos será clave a la hora de gestionar cada tipo de información.

2. Conceptos y características

Concepto de fichero

Un **fichero o archivo** es un conjunto de datos estructurados y ordenados que tienen entre sí una relación lógica y están almacenados en un soporte de información determinado.

Un fichero está compuesto por estructuras de datos más simples denominadas **registros**. Todos los registros de un fichero son del mismo tipo, es decir, un fichero está formado por un conjunto de registros homogéneos. Los registros a su vez están formados por **campos** que contienen información referente a un elemento o característica en particular dentro del fichero.

Registros físicos y lógicos

La palabra “registro” hace alusión a dos conceptos distintos:

Un **registro lógico** es la estructura de datos que corresponde a cada elemento que compone el fichero.

Un **registro físico o bloque** es el conjunto de información que puede ser escrito o leído de una sola vez. Esto dependerá en gran medida de las características físicas de la memoria secundaria que lo almacene y de cómo maneje el dispositivo el sistema operativo.

Se denomina **factor de bloqueo** a la correspondencia entre registros físicos y lógicos. Se pueden dar las siguientes situaciones:

- Registro lógico > Registro físico. En un bloque se contienen varios registros físicos por bloque; se denominan **registros expandidos**.

- Registro lógico = Registro físico. El factor de bloqueo es 1 y se dice que los registros no están bloqueados.
- Registro lógico < Registro físico. El factor de bloqueo es mayor que 1 y los registros se dicen que están bloqueados.

Es importante tener en cuenta este el factor de bloqueo en el diseño del fichero, ya que, a mayor tamaño de bloque, mayor velocidad en los procesos de entrada/salida ya que menor será el número de accesos al dispositivo necesarios. Sin embargo, un tamaño pequeño de bloque mejorará el aprovechamiento del soporte, ya que disminuirá la fragmentación interna del fichero (espacios desocupados).

La **dirección lógica** de un registro es la posición relativa que ocupa en el fichero, mientras que la **dirección física** es la posición real o efectiva donde se encuentra dicho registro en el soporte de información.

Clave

En el fichero, los registros aparecen al usuario en secuencia lógica, es decir, ordenados por su dirección lógica, también llamada **clave** del fichero. No obstante, el orden físico de los registros de un fichero en el disco puede no tener ninguna relación con la información que contiene.

Una **clave** es un campo de datos que identifica el registro y lo diferencia de los otros registros. Esta clave obviamente no se puede repetir en ningún caso y debe ser diferente para cada registro para que pueda ser identificado de forma única.

Un fichero puede estar indexado por varios campos clave para que sea más rápido encontrar cada registro; en este caso una de ellas debe ser la **clave primaria** o clave principal y las demás se denominan claves secundarias.

Una clave, así mismo, puede estar compuesta por más de un campo, denominándose en este caso **clave compuesta**.

El sistema operativo es el encargado de realizar la transformación de la dirección lógica usada en los programas a la dirección física con la que se direcciona el soporte.

Dispositivos secuenciales y de acceso directo

Los dispositivos o soportes de memoria auxiliar pueden ser de dos tipos: **secuencial** o **no direccionables**, y de **acceso directo** o **direccionables**. En los soportes no direccionables, si se quiere acceder al registro n hay que leer los n-1 registros anteriores. En los soportes direccionables, por el contrario, se puede acceder directamente a un registro físico sin más que dar su dirección física, sin necesidad de recorrer otros registros.

3. Operaciones con ficheros

Los sistemas informáticos incluyen en su software propio una serie de programas de utilidad para efectuar operaciones básicas con ficheros. Dichos programas se utilizan mediante un lenguaje de control. Para explotar mejor las características de los ficheros, existen paquetes de programas específicos, denominados genéricamente **sistemas de gestión de ficheros**, que permiten diseñar fichero con determinadas estructuras y realizar recuperaciones y actualizaciones de una forma cómoda y eficaz.

Esta gestión debe ser tenida en cuenta en el diseño del fichero y en su programación. A continuación, se listan las operaciones posibles, aunque no siempre es necesario implementarlas todas:

Creación. Para poder realizar cualquier operación sobre un fichero es necesario que exista previamente. La creación física de un fichero corre a cargo del sistema operativo. En el momento de la creación puede añadirse la información requerida para su posterior tratamiento, o bien puede crearse vado para que otros procesos hagan esta tarea.

Apertura. El sistema operativo crea un dispositivo lógico temporal que tiene asociados un identificador y un buffer de E/S. Antes de realizar cualquier operación sobre un fichero, es necesario abrirlo para que el sistema operativo relacione el dispositivo lógico con el fichero físico, y para indicarle el tipo de operaciones que vamos a realizar, ya sean de consulta, modificación o ambas.

Cierre. Una vez finalizadas las operaciones efectuadas sobre el fichero, es necesario cerrarlo para destruir las estructuras que se han creado para su manipulación, tanto del programa como del sistema operativo. Esta operación actualiza el fichero, escribiendo toda la información que pudiera quedar en el buffer de E/S.

Borrado o destrucción. Elimina la información del fichero, así como su estructura. La destrucción del fichero supone el fin de su existencia.

Ordenación. Consiste en disponer lógicamente todos los registros de un fichero según los valores crecientes o decrecientes de uno o varios de sus campos.

Fusión. Permite obtener a partir de varios ficheros ordenados por un mismo campo y con la misma estructura, otro fichero que contenga todos los registros de ambos y que se mantenga ordenado por el mismo campo.

Intersección. Consiste en crear un nuevo fichero a partir de campos comunes de dos o más ficheros. El fichero resultado y los ficheros origen pueden tener la misma o distinta estructura.

Partición. Permite obtener varios ficheros a partir de un único fichero, de acuerdo con alguna condición que deben cumplir los registros.

Compactación. Tiene como fin reorganizar físicamente los registros de un fichero para eliminar los huecos libres que se producen por la eliminación de registros (borrado lógico, se ve más adelante).

Copia. A partir de un fichero origen, consiste en crear un nuevo fichero con la misma estructura y el mismo contenido que el primero, dejando intacto el fichero original. Es una operación recomendable como medida de seguridad.

Consulta o recuperación: Esta operación se realiza a nivel de registro para conocer la información contenida en el fichero

Mantenimiento o actualización: Una vez que el fichero está creado, puede surgir la necesidad de modificar la información que contiene. Esto supone tres tipos de operaciones distintas a nivel de registro:

- **Inserción** de un registro cuando aparezcan nuevas entidades (admisión de un nuevo alumno en el Instituto).
- **Modificación** de un registro para el que se han producido cambios. Consiste en el cambio de uno o varios campos del registro (modificar la dirección de un alumno que ha cambiado de vivienda).
- **Eliminación** o *borrado* de un registro porque hayan desaparecido las correspondientes entidades (alumnos que se han dado de baja por cambio de Instituto o por haber terminado sus estudios). La supresión de un registro se puede realizar de dos modos:
 - **Borrado lógico**, que consiste en la modificación del valor de un campo que el programa interpretará como que el registro puede ser eliminado en la próxima operación de compactación o si fuese necesario reescribirlo.
 - **Eliminación real**, que consiste en hacer que ese registro sea inaccesible, o bien ocupar su espacio con otros registros de ese fichero. Este tipo de borrado suele ser costoso.

4. Tipos de ficheros

Los ficheros se pueden clasificar según varios criterios:

4.1. Segundo su función.

4.1.1. Permanentes.

Contiene información relevante para una aplicación y su funcionamiento. Generalmente no pueden crearse a partir de otros ficheros. A su vez se clasifican en:

- **Maestros.** Contienen el estado actual de los datos susceptibles de ser modificados. Todos los procesos están orientados a actualizar los ficheros maestros o a obtener datos de ellos.
- **Constantes.** Es aquel que contiene datos fijos para la aplicación, de sólo lectura.
- **Históricos.** Contiene datos que fueron actuales en un tiempo anterior, se conservan para reconstruir la situación actual.

4.1.2. Temporales.

Contiene información relevante para un determinado proceso de la aplicación. Se genera a partir de los datos de ficheros permanentes y su vida es corta. Pueden ser:

- **Intermedios.** Se utilizan para almacenar los resultados de un programa que han de ser utilizados por otro dentro de una misma tarea.

- **De maniobras.** Se utilizan para almacenar datos propios de un programa que éste no puede conservar en memoria principal.
- **De resultados.** Usados para almacenar datos elaborados que van a ser transferidos a un dispositivo de salida.

4.2. Segundo criterio de clasificación: Según su contenido.

4.2.1. Ficheros de texto

Consisten en sucesiones de caracteres producto de la conversión del formato interno de los datos a un código alfanumérico. Son de acceso secuencial y están organizados en registros de longitud indefinida separados por un carácter especial de fin de línea. No hay una relación uno a uno entre los datos existentes en la memoria y los que aparecen en el soporte.

4.2.2. Ficheros binarios

Son una imagen del formato interno de los datos en la memoria. Son de acceso directo, y se organizan en registros de longitud fija o en secuencias de bytes.

4.3. Tercer criterio de clasificación: Según la longitud de los registros.

Los registros que componen un fichero pueden o no tener toda la misma longitud. Pueden existir campos de longitud variable o pueden contener campos que se repiten un número variable de veces. Los ficheros por tanto pueden ser de los siguientes tipos, según sus registros:

- **Longitud fija.** La suma de los caracteres de todos los campos de los registros es constante. Todos los registros del fichero tendrán la misma longitud.
- **Longitud variable.** La longitud de los registros varía, oscilando entre un mínimo y un máximo, siendo pequeño el número de caracteres en que difiere la longitud. En estos casos, el sistema reserva una palabra al comienzo de cada registro para anotar su longitud.
- **Delimitados.** La longitud del registro es variable y no es posible conocer en qué medida difieren unos registros de otros. El sistema incluye un **carácter especial** para indicar el final del registro. En este caso, se dice que el fichero es de tipo texto.
- **Indefinido.** Su longitud es totalmente variable. En este caso, el sistema operativo no realiza ninguna gestión sobre la longitud de los registros del fichero. El programa de usuario es el que se encarga de localizar el principio y el final de cada registro.

4.4. Cuarto criterio de clasificación: Según el tipo de acceso

- **Secuencial.** Para acceder a un registro es necesario recorrer uno a uno por orden todos los registros partiendo del primero. Es el único modo de acceso en los ficheros de asignación contigua y enlazada, y es una posibilidad en los ficheros de asignación indexada.
- **Directo.** Es posible acceder a sus registros o bytes en cualquier orden, tanto en escritura como en lectura. Sólo lo permiten los ficheros de asignación indexada.

5. Organización de ficheros

El ordenador debe tener acceso a los ficheros creados por los usuarios, ya sea para tomar de ellos alguna información o para grabarla. El acceso a un fichero está íntimamente ligado a la organización de dicho fichero. La organización del fichero indica cómo están dispuestos los registros en el soporte material con objeto de conseguir su utilización más eficiente.

Cuando se crea un fichero, es necesario especificar qué organización tendrá, ya que, esta organización va a determinar el tipo de acceso que podemos utilizar. Existen dos tipos de accesos: *acceso secuencial*, en el que para acceder a un determinado registro n hay que recorrer sucesivamente los n-1 registros anteriores en la misma secuencia en que fueron escritos, hasta encontrar el registro adecuado, y *acceso directo*, en el que dada una llave se permite acceder directamente al registro sin necesidad de recorrer todos los anteriores.

Los tipos de organización de ficheros son básicamente cuatro: organización secuencial, organización secuencial encadenada, organización secuencial indexada, y organización directa o aleatoria.

5.1. Organización secuencial

En este tipo de organización, los registros se encuentran almacenados de forma contigua, siguiendo la secuencia lógica del fichero. Todas las operaciones que se realizan sobre el fichero se hacen según esta secuencia. Esta es la única organización de ficheros susceptible de ser gestionada en un dispositivo no direccionable (soportes secuenciales, tales como una cinta magnética o la banda magnética de una tarjeta de crédito).

Las distintas operaciones que se pueden realizar sobre ficheros con esta organización son:

- **Añadir.** Sólo es posible escribir al final del fichero. Los registros se van almacenando uno a continuación del otro, en el orden en que se desea que estén en el fichero.
- **Consulta o recuperación.** Se realiza en orden secuencial, es decir, para leer el registro que ocupa la posición n es necesario leer previamente los n-1 anteriores.
- **Operaciones de actualización:** **inserción, modificación y eliminación.** Estas operaciones se realizarán en un fichero de movimientos independiente (también secuencial). El fichero de movimientos almacenará las modificaciones a realizar sobre el fichero, y en un momento determinado se llevarán a cabo mediante el borrado de los dos ficheros antiguos y la escritura de su información en uno nuevo.

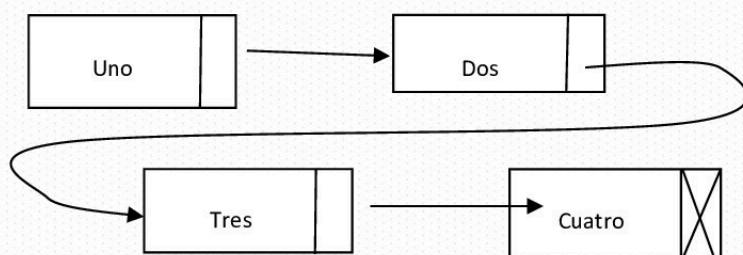
Las principales características de esta organización son:

- Aprovecha al máximo el soporte, al no dejar huecos entre los registros, y se puede grabar en cualquier tipo de soporte, tanto secuencial como direccionable.
- El acceso al registro siguiente es muy rápido, por lo que se hace ideal cuando en cada operación de actualización o consulta se van a procesar la mayoría de los registros.
- El único modo de acceso es el acceso secuencial. Esto hace que este tipo de organización no sea adecuado para ficheros en los que se necesita procesar frecuentemente registros aislados.

5.2. Organización secuencial encadenada

Si se utilizan dispositivos direccionables para un fichero secuencial tenemos lo que se denomina fichero secuencial “encadenado”.

Los registros que componen un fichero con organización secuencial encadenada almacenan, además de su propia información, un puntero (tipo de dato que almacena una dirección de memoria) con la dirección del registro siguiente, según el orden lógico del fichero. Desde el punto de vista lógico, el fichero está ordenado según el valor de alguna llave. Los registros se encuentran colocados en direcciones físicas totalmente arbitrarias, pero los punteros permiten asegurar la secuencia lógica del fichero.



Con esta organización, se pueden realizar las siguientes operaciones:

- **Añadir.** Para añadir un registro al final del fichero, se localiza la posición del último registro del fichero. El puntero de este último registro contendrá una dirección nula. Una vez localizado el final del fichero, el nuevo registro se escribirá en una zona libre, colocando en su campo de puntero una dirección nula. Finalmente, se modifica el último registro para actualizar el valor de su puntero, de forma que contenga la dirección del nuevo registro.
- **Recuperación o consulta.** La consulta es secuencial. Para leer un determinado registro, se accede al primero en la lista y se comprueba si es el registro buscado. Si no es así, se lee la dirección del siguiente (gracias al puntero) y se accede a dicha dirección. El proceso continúa hasta que se localiza el registro deseado o se llega al final del fichero.
- **Inserción.** Para insertar un registro, es necesario localizar la posición en que se desea insertar, es decir, entre qué dos registros se quieren situar (registro anterior y posterior). Físicamente, el registro se escribe en una dirección de memoria arbitraria que se encuentre disponible, colocando en su campo de puntero la dirección del registro al cual va a preceder. Luego se modifica el registro anterior para actualizar el valor de su puntero, de forma que contenga la dirección del nuevo registro. El añadir un registro es un caso particular de la inserción, en el que se desea insertar un registro al final del fichero.
- **Modificación.** Si la modificación no implica un aumento en la longitud ni una alteración del valor del campo llave, se localizará el registro y simplemente habrá que sobrescribir en la misma posición. En caso contrario, primero se insertará un nuevo registro que incluya la modificación, y posteriormente se borrará el registro con la información desactualizada.

- **Borrado.** Para eliminar un registro del fichero, sólo hay que destruir su dirección del puntero del registro anterior, copiando en dicho puntero la dirección del registro siguiente al que se desea borrar, es decir, la dirección que contiene el puntero del registro a eliminar.

5.3. Organización secuencial indexada

Un fichero con organización secuencial indexada se considera compuesto por dos estructuras o zonas:

- **Zona de registros.** Contiene todos los registros del fichero, ordenados según el valor de una llave. Su estructura es semejante a la de un fichero con organización aleatoria y direccionamiento directo o a un fichero secuencial puro en el que se pueda direccionar a nivel de registro.
- **Zona de índices.** Se procesa sólo de forma secuencial. Su estructura es la de un fichero secuencial puro en que cada registro contiene dos campos: un campo *llave* (que contendrá algunos valores de la llave del fichero) y un campo *dirección* (que contendrá la dirección de un registro del fichero).

La zona de registros está dividida en tramos lógicos. Cada tramo está formado por una serie de registros consecutivos. Por cada tramo de la zona de registros hay un registro en la zona de índices. En este registro se encuentra el valor mayor de llave del tramo (valor de llave del último registro del tramo) y la dirección del primer registro del tramo. Ambas zonas pueden o no estar en un mismo fichero del sistema. En cualquier caso, la gestión de la estructura la realiza el sistema operativo o un paquete software especial. Por tanto, el usuario de esta estructura no necesita conocer la existencia de ambas zonas, pudiendo contemplarlas como un todo.

ZONA DE ÍNDICES

Llave	Dirección
Cáceres	1
Logroño	4
Oviedo	8
Zaragoza	10

ZONA DE REGISTROS

1	Almería
2	Álava
3	Cáceres
4	Cuenca
5	Huesca
6	Jaén
7	Logroño
8	Madrid
9	Oviedo
10	Zaragoza

Con ficheros secuenciales indexados se pueden realizar las siguientes operaciones a nivel de registro:

- **Consulta.** Se pueden realizar consultas secuenciales, como en los ficheros con organización secuencial. Además, se pueden realizar consultas por llave (localizar un registro conocida su llave) sin necesidad de leer los registros que le anteceden en el fichero. Para realizar una consulta por llave se utiliza el siguiente proceso:
 - 1). Leer secuencialmente las llaves en la zona de índices hasta encontrar una mayor o igual a la del registro buscado. Obtener la dirección de comienzo del tramo donde se encuentra el registro.
 - 2). Leer secuencialmente en la zona de registros, a partir de la dirección obtenida en la zona de índices hasta encontrar el registro buscado o uno con valor de llave mayor que el buscado. En este último caso el registro no se encuentra en el fichero.
- **Inserción.** Dado que ambas zonas son secuenciales, no es posible insertar un registro en ficheros con esta organización. En algunos casos, se permite la escritura de nuevos registros al final del fichero. Estos registros, como es lógico, no podrán ser consultados por llave con el procedimiento antes descrito.
- **Eliminación.** No es posible borrar un registro, aunque sí se permite un borrado lógico (marcándolo).
- **Modificación.** Las modificaciones son posibles tan solo si el registro no aumenta de longitud al modificarlo ni supone una variación del valor de la llave del registro.

Esta organización resulta especialmente útil cuando se deben combinar consultas a registros concretos y el procesamiento secuencial de todo el fichero. Su principal inconveniente es la imposibilidad de introducir nuevos registros o actualizaciones en los registros ya existentes en el fichero sin volver a realizar una *reorganización* del mismo con las modificaciones efectuadas.

Para permitir actualizaciones es necesario incluir en la estructura una **zona de desbordamientos** (o zona de *Overflow*). En la zona de desbordamientos los registros están desordenados, cada nuevo registro se escribe al final de ésta sin tener en cuenta el orden de la llave. Esto complica la búsqueda por llave, pues, si el registro no es encontrado en la zona de registros, es necesario buscarlo secuencialmente en la zona de desbordamientos. Además, se imposibilita la consulta secuencial del fichero, ya que, los registros no aparecen ordenados por llave.

Otra mejora de la estructura anterior es incluir punteros entre los registros, de forma que éstos mantengan el orden lógico de los registros. A esta organización se le llama **secuencial indexada encadenada**. La estructura de un fichero será similar a la de un fichero secuencial indexado, salvo que se ha previsto un campo en cada registro para albergar un puntero.

Para insertar un nuevo registro, en este caso, es necesario encontrar el que le sigue en la zona de registros. Se escribe el nuevo registro en la zona de desbordamientos y se reescribe el registro encontrado como siguiente según el orden lógico, para incluir el puntero al registro recién grabado.

La eliminación de un registro debe realizarse por marcas. Esto no implica en ningún caso la realización de modificaciones en el índice, pero degrada el fichero. La consulta es semejante a la realizada para el caso no encadenado.

5.4. Organización directa o aleatoria

No tiene por qué existir ninguna relación lógica entre los registros y su colocación física. Cada registro se sitúa en una dirección de memoria que se calcula para cada uno de ellos aplicando una fórmula o algoritmo matemático.

Existen muchos métodos para generar las direcciones de los registros. Todos ellos tomarán el valor de un campo del registro, aplicarán la transformación y obtendrán la dirección. Por tanto, dependiendo del fichero concreto, se elegirá el método que asegure que las direcciones estarán dentro del rango permitido intentando distribuirlas de modo que existan pocos **sinónimos** (registros con diferentes llaves, pero con una misma dirección física). Cuando se detecta que la dirección asignada a un registro está ya ocupada por otro, se puede optar por dos alternativas:

- Buscar una nueva dirección libre en el mismo espacio de fichero hasta encontrar una posición libre donde almacenar la información.
- Reservar una zona de desbordamiento e ir almacenando los sinónimos en esta zona de manera consecutiva a medida que van apareciendo.

El problema principal de esta organización es la elección de la función de transformación o **método de direccionamiento** que se ha de usar para cada tipo de fichero. Existen distintos algoritmos para determinar la dirección a partir de una llave:

- **Direccionamiento directo.** Este método consiste en asignar a cada llave una dirección lógica, es decir, la transformación asigna como dirección el valor de la llave. El problema principal radica en que tiene que haber tantas direcciones como valores de la llave, y que las llaves han de ser numéricas. La ventaja es que no se producen sinónimos y el fichero queda ordenado.
- **Direccionamiento asociado.** Normalmente se utiliza cuando las llaves son alfanuméricas, y consiste en asociar a cada llave una dirección lógica por medio de una tabla. Se construye una tabla con todas las llaves del fichero y la dirección física de cada registro. Mediante una búsqueda secuencial en la tabla de llaves, obtenemos la dirección lógica de almacenamiento. Este tipo de almacenamiento no produce sinónimos, pero necesita espacio adicional para la tabla. Se puede hacer también que las llaves se encuentren ordenadas en la tabla, con lo que la búsqueda sería más rápida.
- **Direccionamiento calculado (hashing).** En este tipo, la dirección de cada registro se obtiene mediante una serie de cálculos matemáticos realizados con la llave. El principal problema que plantea es la existencia de sinónimos, pues distintos cálculos pueden llevar al mismo resultado. Existen varios sistemas de cálculo, entre los que podemos citar:
 - **División por un número primo.** Si N es el número máximo estimado de registros en el fichero y P es el número primo más próximo a N , se divide la llave por P y se utiliza como dirección el resto de la operación.
 - **Cuadrado de la llave.** Si la llave no es un número muy alto, se eleva al cuadrado y se extraen como dirección unas cuantas cifras intermedias
 - **Truncamiento o extracción.** Se trunca la llave, quedándonos con sus 4 o 5 últimas cifras, y a partir de ellas con una serie de cálculos se obtiene la dirección.

- **Plegamiento.** Se utiliza cuando la llave toma valores numéricos muy grandes. Consiste en recortar la llave en trozos y después sumar dichos trozos. Al resultado de la suma se le puede volver a aplicar cualquier método de los descritos.

Las *operaciones básicas* sobre ficheros con esta organización son las siguientes:

- **Consulta.** Se realiza siempre por llave. Para leer un determinado registro, únicamente se ha de aplicar el algoritmo de transformación a la llave. Si el registro buscado no se encuentra en la dirección resultante, se aplicarán los algoritmos de resolución de sinónimos.
- **Inserción.** Para insertar un nuevo registro, se aplica a la llave el algoritmo de transformación elegido. Si la dirección que resulta de aplicar el método de direccionamiento ya está ocupada por otro registro, se emplea un algoritmo de resolución de sinónimos.
- **Modificación.** Siempre se puede realizar esta operación. Para localizar el registro, se aplica la transformación correspondiente a la llave. Una vez que se tiene la dirección, se modifica la información del registro.
- **Borrado.** Siempre se realiza un borrado lógico. Se localiza el registro mediante los algoritmos descritos y se marca con un valor determinado para indicar que no es un registro válido.

La gran flexibilidad y la rapidez de consulta son algunas ventajas de esta organización. Entre los inconvenientes que presenta, destacan el mal aprovechamiento del espacio, la necesidad de utilizar soportes direccionables y la exigencia de una planificación previa para utilizar el método más apropiado.

5.5. Ficheros con organización secuencial indexada encadenada

Otra mejora de la estructura anterior es incluir punteros entre los registros, de forma que éstos mantengan el orden lógico de los registros. A esta organización se le llama **secuencial indexada encadenada**. La estructura de un fichero será similar a la de un fichero secuencial indexado, salvo que se ha previsto un campo en cada registro para albergar un puntero.

Para **insertar** un nuevo registro, en este caso, es necesario encontrar el que le sigue en la zona de registros. Se escribe el nuevo registro en la zona de desbordamientos y se reescribe el registro encontrado como siguiente según el orden lógico, para incluir el puntero al registro recién grabado.

La **eliminación** de un registro debe realizarse por marcas. Esto no implica en ningún caso la realización de modificaciones en el índice, pero degrada el fichero. La **consulta** es semejante a la realizada para el caso no encadenado, se averigua la posición de entrada al fichero utilizando la tabla de índices, y se sigue el encadenamiento de los punteros para localizar el registro deseado.

6. Conclusión

Los datos procesados por las computadoras se organizan en jerarquías de datos formando estructuras a partir de bits, caracteres, campos, etc.

Los campos se agrupan en registros, un fichero, por tanto, está compuesto de registros que son tratados como una unidad de almacenamiento y organizados de forma estructurada para la búsqueda de un dato individual.

La estructura y la organización que tenga el fichero determinará la forma en que podrá ser accedido y las operaciones que se puedan realizar sobre él.

Finalmente, los ficheros pueden formar parte de bases de datos complejas o ser directamente utilizados por programas independientes o por el propio sistema operativo.

6.1. Relación del tema con el sistema educativo actual

Este tema es aplicado en el aula en los módulos profesionales siguientes, con las atribuciones docentes indicadas (PES/SAI):

Grado Superior

- DAW - DAM –Programación (PES)

7. Bibliografía

- De Anasagasti, Miguel. "Fundamentos de la Computadora" 9^aed 2004 Edt. Paraninfo
- Patterson D.A. y Hennessy JL. "Estructura y diseño de computadoras: la interfaz hardware/software" 4^a Ed. (2005) Edt McGraw-Hill
- Prieto A, Lloris A, Torres JC. "Introducción a la Informática" 4^aed. (2006) Edt. McGraw-Hill
- Stallings W. "Organización y Arquitectura de Computadoras" (2006) 5^a Ed. Edt. Prentice-Hall
- Ramos A, Ramos MJ y Viñas S "Montaje y Mantenimiento de Equipos" (2012). Edt McGraw-Hill
- Jiménez Cembreras, Isabel M^a "Sistemas Informáticos" 2^aEd (2018) Edt. Garceta