1. **Archivos Ordenados y NO Ordenados.**

* No ordenados: Es el tipo más simple y básico de organización de archivos. Los archivos se colocan en el archivo en el orden en que son insertados, los nuevos se insertan al final. Se utilizan índices secundarios como camino de acceso adicional. Insertar un registro es muy eficiente, pero buscar un determinado registro es muy costoso ya que se debe recorrer gran parte del archivo.

*\* Beneficios: Inserción y eliminación rápida. Contra: Búsqueda costosa.*

* Ordenados: Estos archivos se ordenan en el disco en base a un campo de ordenación. El campo de ordenación es también llamado campo clave cuando por dicho campo sirve para identifica unívocamente a un registro. No es eficiente si se busca por un campo que no es el de indexación.

*\* Beneficios: Método de acceso y búsqueda de un registro muy rápida. Contra: inserción y eliminación muy costosa.*

1. **¿Qué razones hay para tener registros de longitud variable?**

Las razones para tener registros de longitud variable pueden ser varios:

1. Los registros son todos del mismo tipo pero hay campos con tamaño variable.
2. Los registros son todos del mismo tipo, pero hay campos con múltiples valores en registros individuales.
3. Los registros son todos del mismo tipo pero uno o más de los campos son opcionales.
4. El archivo contiene registros de diferentes tipos por lo que su tamaño es variable.

1. **Analice las distintas técnicas para eliminar registros**

La eliminación de un registro se da en archivos ordenados y no ordenados. En los no ordenados debe encontrarlo, copiar el bloque a un buffer, eliminarlo y reescribir el bloque en el disco. Otra forma es tener un bit de marcador lógico de eliminación evitando de esta forma tener que copiar y re escribir el bloque entero, posterior a esto puede realizarse una pulcra de los archivos eliminando y reorganizando el archivos.

En los ordenados, es una operación muy costosa ya que se debe conservar el orden, se puede utilizar los ya mencionados marcadores de borrado y actualizarlo periódicamente como ya se explico para evitar el coste de la operación cada vez que ocurra.

1. **Definir “Organización de Archivos” y “Métodos de Acceso”**

La **organización de archivos** (ordenado, no ordenado, disperso) se refiere a la organización de los datos de un archivo en registros, bloques y estructuras de acceso; esto incluye la forma en que los registros y los bloques se colocan en el medio de almacenamiento físico y se interrelacionan entre sí.

El **método de acceso** (secuencia, directo), en cambio, consiste en un grupo de programas que permiten la aplicación de operaciones a un archivo (como Find o Locate que busca el primer registro que satisface una condición de búsqueda, FindNext que busca el siguiente registro que cumple la condición, etc.), para obtener búsquedas eficientes. En general es posible aplicar varios métodos de acceso diferentes a una organización de archivo determinada.

1. **Dispersión Interna y externa. Explicar conceptos, ventajas, desventajas.**

Dispersión es otra técnica de organización primaria de archivos que se basa en la dispersión (hashing), proporciona un acceso muy rápido a los registros para búsquedas que cumplan con ciertas condiciones. Dichas condiciones de búsqueda deben realizarse sobre el campo de dispersión, que en caso de que también sea el campo clave del archivo se le llama clave de dispersión. Este proceso se basa en la utilización de una función llamada **función de dispersión** o **función de aleatorización** la cual toma el valor del campo de dispersión y devuelve como salida una dirección de memoria a ser asignada al registro que contiene dicho valor.

1. **Dentro de las técnicas de dispersión se encuentra la “Dispersión Interna”. Explique el concepto.**

Se trata de la dispersión implementada para la organización de archivos internos, la cual maneja un arreglo de registros de 0 a M cuando se tengan M direcciones para utilizar. Uno de los problemas de esta técnica es que ningún algoritmo de hashing garantiza que para distintos valores de entrada siempre devuelva valores distintos de salida.

1. **Dentro de las técnicas de dispersión se encuentra la “Dispersión Externa”. Explique el concepto.**

La dispersión de archivos en disco se denomina dispersión externa. Esto surge de la necesidad de adecuarlo a las características del almacenamiento en disco, el espacio de direcciones destino se divide en cubetas, cada una de las cuales contiene varios registros. Cada cubeta es un bloque de disco o bien un grupo de bloques contiguos. La función de dispersión establece una transformación entre la clave y un número de cubeta relativo, en vez de asignar una dirección de bloque absoluta a la cubeta. El problema de las colisiones es menos grave cuando se usan cubetas porque tanto registros como quepan en una cubeta podrán dispersarse en la misma sin causar problemas. En caso de que se llene una cubeta y se necesite insertar un nuevo registro se puede mantener una variación de la técnica de encadenamiento manteniendo una cubeta de desbordamiento y apuntadores de registro para cuando se den estos casos.

1. **¿Qué es una colisión? ¿Que técnicas hay para solucionarlas? Explicarlas.**

Respuesta:

* + Colisión: Una colisión hace referencia a cuando la función de hashing para un distinto valor de entrada devuelve una misma salida (dirección de memoria), es decir que cuando se intenta guardar un registros al ir a la ubicación asignada se encuentra con que ya existe un registros en la misma. El proceso de encontrar una nueva ubicación se lo llama resolución de colisiones.
  + Resolución de colisiones: Existen varios métodos para resolver las colisiones, entre ellos podemos encontrar los siguientes:
    - Direccionamiento Abierto: Partiendo de la dirección de memoria ocupada que causo la colision, se analizan las siguientes hasta encontrar una libre, esta le es asignada al registro a insertar.
    - Encadenamiento: Cuando se usa este método se mantienen áreas de desborde, por lo regular mediante la extensión del arreglo con varias posiciones de desborde. A demás se agrega un puntero a cada registro. Las colisiones las resuelve ubicando el registro que causo la colisión en una dirección libre en el área de desbordamiento y apuntando el registro con el puntero del registro que ya se encontraba en la dirección ocupada.
    - Dispersión Múltiple: El programa aplica una segunda función de dispersión si la primera produce una colisión. Si resulta otra el programa usa direccionamiento abierto o aplica una tercera función de dispersión y luego utiliza direccionamiento abierto si es necesario.

1. **¿Cuáles son las técnicas de dispersión dinámica? Describa su funcionamiento.**

* Dispersión dinámica: Con técnica el número de cubetas no es fijo, sino que aumenta o disminuye según las necesidades. Comienza con una cubeta y una vez se desborda la misma se divide en dos y así sucesivamente.
* Dispersión Extensible: Se basa en un arreglo de 2 a la d donde d es la profundidad global. Si d=3 entonces bloque de 3 bits.
* Lineal: Aumenta o disminuye el archivo de dispersión sin necesidad de u directorio. Se van dividiendo las cubetas en orden lineal a medida que se provoque un desborde.

1. **Como mejorar la eficiencia de las búsquedas en un archivo de índices con la indización de múltiples niveles.**

Para mejorar la eficiencia de las búsquedas en un archivo de índices con la indización de múltiples niveles se pueden generar más niveles, donde los primeros niveles sean no densos, minimizando la cantidad de entradas del bloque y en la última sea del tipo denso, posibilitando entonces determinar la existencia o no de un registro en su búsqueda sin necesidad de acceder a la tabla.

1. **Definir índices primarios, de agrupamiento y secundarios. ¿Cuales son densos y cuáles no? Con influyen las diferencias entre ellos a la hora de ordenar o hacer una búsqueda.**

* Índices Primarios (No denso o Disperso): Se construyen sobre el campo clave de ordenamiento de un archivo ordenado de registros.
* Índices Agrupados (No denso o Disperso): Se construyen sobre un campo de ordenamiento el cual abarca varios registros con el mismo valor.
* Índices Secundarios (Denso): Se construyen sobre un campo por el cual no se encuentra ni se utiliza para ordenar el archivo indexado.

Índices Densos: Son aquellos que poseen una entrada por cada entrada del archivo referenciado.

\*Tipos de Índices por estructura de referencia:

- Longitud Fija por **puntero a bloque**.

- Longitud Variable por **lista invertida**.

- Longitud Fija por **mapa de bits**.

1. **Definir REUNION del algebra relacional. ¿Porque otras operaciones se pueden reemplazar?**

Se trata de un operador del Algebra Relacional el cual relaciona 2 tuplas de relaciones distintas las cuales deben compartir un mismo atributo (y mismo nombre) y a su vez en dichas tuplas para ese atributo poseer el mismo valor.

Dicho operador se podría reemplazar por la combinación del operador rename, el operador times y finalmente el where.

1. **¿Qué propiedades posee el Algebra Relacional? ¿Cuáles son los operadores tradicionales?**

El algebra relacional posee 3 propiedades de herencia:

1. Clausura: El resultado de una operación entre relaciones, es otra relación. Esto permite la creación de expresiones con operadores anidados. Para que las relaciones resultantes de las operaciones intermedias funciones como relaciones y puedan ser tomadas como entrada para el resto de los operadores.
2. Nombre de Atributos: Los nombres de atributos de las tablas originales deben formar parte de la cabecera de la relación resultante.
3. Claves Candidatas: Las operaciones realizadas entre diferentes relaciones no deben modificar propiedades de los atributos que forman una clave candidata.

Operadores Tradicionales: Union, Minus, Intersects, Times, Divide By, {}, Restriction, Join.

Operadores Especiales: Operadores de Totales (Count, Max, Min, AVG, Sum), Extend, Summarize.

1. **¿Cuáles son los aspectos de los que se encarga el modelo relacional?**

* Garantiza herramientas para evitar la duplicidad de registros, a través de campos claves o llaves.
* Garantiza la integridad referencial: Así al eliminar un registro elimina todos los registros relacionados dependientes.
* Favorece la normalización por ser más comprensible y aplicable.

1. **¿Cuáles son las propiedades de las Relaciones?**

* No existe, ni se puede realizar un orden de las tuplas ya que las relaciones se comportan como conjuntos y los conjuntos no pueden ordenarse. (De arriba hacia abajo)
* Los atributos están en desorden lógico, los mismos se encuentran de Izquierda a Derecha.
* No existen tuplas duplicadas ya que se trata de un conjunto de tuplas.
* Cada tupla contiene exactamente un valor para cada atributo.

1. **Explique qué es y para qué sirve la propiedad de cierre del Algebra Relacional. ¿Qué relación tiene dicha propiedad sobre las cabeceras de las relaciones?**

La propiedad de cierre garantiza que el resultado de cualquier operación que se realice entre 2 relaciones mediante el Algebra Relacional da como resultado una nueva relación que cumple con todas las propiedades de las mismas. Gracias a esta propiedad es que se pueden anidar operaciones del Algebra Relacional para obtener la relación resultado buscada.

Con respecto a la cabecera de las relaciones, dicha propiedad garantiza también que tanto los atributos de la cabecera como el dominio de cada uno de ellos mantendrán sus características.

1. **¿Qué son los archivos ISAM? Describa brevemente su funcionamiento. ¿Es una estructura fácil de migrar? ¿Por qué?**

Un archivo ISAM (Indexed Sequential Access Method o Método de Acceso Secuencial Indexado) se trata de un método el cual se utiliza para almacenar información a la cual se pueda acceder rápidamente.

La información se organiza en ficheros secuenciales los cuales poseen campos de longitud fija. En paralelo se mantiene un conjunto secundario de ficheros dispersos (archivos Hash) los cuales son conocidos como índices que contienen punteros a los registros del archivo secuencia permitiendo acceder directamente a los buscados sin necesidad de recorrer todo el archivo.

**No es fácil de migrar** ya que cada máquina cliente tiene un registro de los punteros a los archivos que utiliza.

1. **Defina el modelo de Red**

Si lo comparamos con el modelo Jerárquico, en lugar de ser un Árbol, se trata de una red interconectada de nodos ya que a diferencia de esta, se puede ingresar por cualquier registro y no necesariamente por el más alto de la jerarquía. Se cambia el concepto de Padre-Hijo por el de Dueño-miembro. Puede haber más de un registro con más de un Dueño y cada registro posee su cadena de apuntadores.

La navegación se produce registro a registro, al ir navegando registro a registro no se puede mantener toda la estructura en memoria, por lo que se debe ir guardando parcialmente la información (Búsqueda y Recuperación).

1. **¿Cuáles son los tipos de conjuntos presentes en el modelo de Red? Indique cuales son las propiedades básicas y cuáles son los tipos especiales de conjuntos.**

Las propiedades básicas de un tipo de conjunto son un NOMBRE (para el tipo de conjunto), un **PROPIETARIO** y un **MIEMBRO**.

Los tipos especiales de conjuntos son: propiedad del **SISTEMA** (no tiene registro propietario, el propietario es el sistema), **MULTIMIEMBRO** (los conjuntos miembro de un registro pueden pertenecer a mas de un tipo de registro) y **RECURSIVOS** (el mismo tipo de registro desempeña tanto un papel de propietario como de miembro.

1. **¿Cuáles son las opciones de ordenamiento de conjuntos? Explíquelos.**

* **Según un campo de ordenamiento**: Los valores de uno o mas campos sirven para ordenar los registros dentro de cada ocurrencia de conjunto en orden ascendente o descendente.
* **Orden por omisión del sistema:** La inserción es arbitraria determinada por el sistema.
* **Primero o Último**: Es equivalente a tener los registros en orden cronológico.
* **Siguiente o Previo:** El nuevo registro miembro se insertara antes o después del registro actual de la ocurrencia de conjunto.

1. **¿Qué son los indicadores de actualidad? ¿Para qué se usan? Enumera cada uno de ellos.**

Son indicadores que se utilizan en las búsquedas dentro de las bases de datos de red, los mismos resultan cruciales para el SGBD para mantenerse al tanto de que registros y ocurrencias de conjuntos tuvo acceso más recientemente.

Existen varios tipos de indicadores de actualidad

* Actual de tipo de registros: Por cada tipo de registros, el SGBD sigue la pista al último registro de ese tipo al que se tuvo acceso. Si todavía no se ha leído ningún registro de ese tipo, el registro actual no está definido.
* Actual de tipo de conjunto: Por cada tipo de conjuntos en el esquema, el SGBD sigue la pista a la última ocurrencia de conjunto de ese tipo de conjuntos a la que se tuvo acceso.
* Actual de unidad de ejecución (CRU: *Current Ofrun Unit*): Es un registro sobre el ultimo al que se tuvo acceso por cada unidad de ejecución, este indicador no discrimina entre tipos de de registros ni conjuntos. Mantiene el último

1. **¿Cuáles son las 3 opciones (restricciones) de retención en conjuntos del modelo de Red? Describa cada una de ellas.**

OPTIONAL (Opcional): Un registro puede desconectarse de su propietario y quedar suelta. Puede subsistir por si solo sin necesidad de formar parte de ningún conjunto.

MANDATORY (Obligatoria): Ningún registro miembro puede existir por sí solo, odrá cambiar de propietario pero no podrá quedarse sin ninguno. Si se define el miembro como OPTIONAL, puede desconectarse de su propietario y quedar suelta.

FIXED (Fija): No permite que el registro quede suelto ni que tampoco cambie de ocurrencia de SET, la restricción es más fuerte que MANDATORY, y estaba contemplada en la mayoría de los SGBD comerciales.

1. **Defina el modelo Jerárquico**

Son bases de datos las cuales almacenan su información en una estructura jerárquica. En este modelo los datos se organizan de forma similar a un árbol visto al revés, en donde un nodo padre de información puede tener varios hijos. El nodo que no tiene padre es llamado **nodo raíz** y los nodos que no tienen hijos se los conoce como **nodos hojas**.

\* Una de las principales limitaciones de este modelo es su incapacidad para representar eficientemente la información sin redundancia de datos.

1. **Enumere las restricciones de integridad en el modelo Jerárquico.**

* Ninguna ocurrencia de registro, excepto la raíz, puede existir si no está relacionado con una ocurrencia de registro padre.
* Ningún registro hijo puede insertarse si no está enlazado a un registro padre.
* Un registro hijo puede eliminarse independientemente de su padre.
* Las reglas anteriores no aplican a registros hijos y padre virtuales,
* Si un registro hijo tiene dos o más registros padre del mismo tipo, este de duplicarse por cada uno.
* Un registro hijo que tenga dos o más registros padre de diferente tipo solo puede tener un padre real, todos los demás deberán representarse como padres virtuales.

1. **¿Qué es un registro Virtual? ¿En qué casos necesario su uso? ¿Por qué?**

Dado que en las relaciones muchos a muchos existe demasiada repetición de datos, se maneja el concepto de registro virtual. Un registro virtual es aquel que no se escribe físicamente en el medio, sino que es una referencia a un registro existente en forma previa. La razón de utilizar registros virtuales es evitar la repetición de datos, lo cual traería consigo los siguientes inconvenientes: redundancia, y alto riesgo de inconsistencias.

1. **¿Qué diferencias hay entre un árbol B y un árbol B+?**

Un árbol-B+ es una variación de un árbol-B.

En un árbol-B+, en contraste respecto un árbol-B, toda la información se guarda en las hojas. Los nodos internos sólo contienen claves y punteros. Todas las hojas se encuentran en el más bajo nivel. Los nodos hoja se encuentran unidos entre sí como una lista enlazada para permitir búsqueda secuencial (más rápido).

Los árboles B+ tienen además punteros extras entre vértices que son “hermanos” para ofrecer más posibilidades simples para mover claves al buscar, insertar y eliminar claves. Estos árboles también permiten lograr índices con menos niveles y más alta capacidad.

1. **¿Cuáles son las técnicas de asignación de bloques de archivo en disco? Describa brevemente cada una de ellas.**

* Asignación Contigua: Como el nombre lo dice, los bloques que pertenecen a un mismo archivo se ubican de manera contigua. Lo que es particularmente útil en el caso de discos mecánicos ya que no es necesario realizar grandes movimientos en el cabeza para leer el archivo. También reduce las búsquedas en el disco, ya que sabe que están en cierto espacio contiguo y no es necesario explorar todo para llegar a la información deseada. Cada directorio contiene para cada archivo la dirección del bloque en que comienza y la longitud del área asignada a este archivo.
* Asignación Enlazada: Cada archivo es una lista ligada de bloques en disco, en cada bloque existe un puntero al bloque siguiente, el resto del bloque es usado para almacenar datos. De esta forma todos los bloques del disco puede ser usados.
* Asignación Indexada: También conocido como asignación por “I-Nodes”, este método lleva el rastro de que bloque pertenece a cada archivo, asignando un I-Node con todos los punteros hacia los demás bloques en el orden correspondiente. Para cada archivo existe un I-Node el cual posee la lista de atributos y las direcciones de bloques de un Archivo.

1. **¿Por qué resulta costoso tener acceso a un bloque en disco?**

Resulta costoso tener acceso a un bloque en disco ya que implica una sucesión de movimientos mecánicos dentro del disco los cuales no son exactamente precisos y a veces para localizarse en una determinada ubicación requieren más de un movimiento (Movimiento de cabeza, esperar posición, leer datos).

1. **¿Qué son los descriptores de archivos? ¿En general como están compuestos?**

Un descriptor de archivo o cabecera de archivo contiene información relativa al archivo que es necesaria para los programas que tienen acceso a los registros del mismo. El descriptor contiene información para determinar las direcciones en disco de los bloques que lo compone, también contiene las descripciones, formatos, orden y longitudes de los campos que lo componen. También informan si se trata de campos de longitud fija y los separadores de campos.

1. **Defina los conceptos de “predicado” y “proposición verdadera”. ¿Qué representan cada uno de ellos? ¿Son conceptos equivalentes?**

* Predicado: En lógica, lo que se afirma o se niega de un sujeto en una proposición.
* Proposición Verdadera: ¿?

Se puede considerar al encabezado de una relación como un predicado y a las tuplas de esa relación como proposiciones verdaderas. En el modelo relacional, en las tablas, los datos son representados por las filas y a su vez estas filas pueden interpretarse como proposiciones verdaderas a las cuales se les pueden aplicar operadores los cuales también van a devolver como resultado proposiciones verdaderas (filas).

1. **¿Qué es y para qué sirve la clave foránea?**

Una clave foránea es un mecanismo de INTEGRIDAD que se establece entre dos relaciones, donde los valores de un subconjunto de atributos de la cabecera de la primera relación, deben coincidir con los valores de la clave primaria de la segunda relación. A la primera relación se la denomina relación referencial y a la segunda relación referida.

1. **Defina independencia lógica y física. Beneficios.**

* Independencia Lógica: Es la capacidad de modificar el esquema conceptual sin tener que alterar los esquemas externos ni los programas de aplicación. Se puede modificar el esquema conceptual para ampliar la base de datos o para reducirlas.

*\* Son más difíciles de lograr que las independencias físicas, ya que los programas de aplicación son fuertemente dependientes de la estructura lógica de los datos a los que acceden.*

* Independencia Física: Es la capacidad de modificar el esquema interno sin tener que alterar el esquema conceptual (o los externos). Se refiere solo a la separación entre las aplicaciones y las estructuras físicas de almacenamiento de los datos.

1. **Diferencias entre administrador de Base de Datos y Administrador de Datos.**

El administrador de base de datos es el encargado de tomar las decisiones y las políticas de lo que se va a hacer con los datos (tiene una visión global del negocio), en cambio el administrador de la base de datos es el que implementa las políticas definidas por el administrador de datos, otorgando permisos, creando estructuras, entre otras cosas.

*\* La principal diferencia del DBA y el DA es que* ***el DA define Políticas*** *y* ***el DBA implementa Políticas****.*

1. **¿Cuáles son los distintos tipos de usuarios de bases de datos existentes? ¿El Administrador de Base de Datos es uno de ellos? ¿Cuál es su función?**

Existen 3 (tres) grandes grupos:

* Programadores: Son aquellos que realizan los programas y se comunican realizando llamadas al DBMS correspondientes, les interesan las estructuras.
* Usuarios Finales: Son aquellos que interactúan con la base desde las terminales. A partir del modelo de Red comenzaron a aparecer los usuarios finales como personas.
* Administrador de Base de Datos (DBA):
  + - Definición del esquema conceptual.
    - Definición del esquema interno.
    - Establecer disponibilidad con los usuarios.
    - Definir restricciones de seguridad e integridad.
    - Definir políticas de backup y restauración.
    - Supervisar el rendimiento.

1. **Características del sistema de sección frontal y sección posterior.**

Sección Frontal del DBMS: Son utilitarios que sirven para cubrir funcionalidades especificas. Dichos utilitarios de la sección frontal están destinados a usuarios finales y programadores.

* + - * Utilitario de reportes/gráficos.
      * Utilitarios generadores de código.
      * Herramientas CASE.

Sección Posterior del DBMS: Son utilitarios que sirven como herramientas para atender problemas, evaluar o controlar el funcionamiento. Por ejemplo supervisar la performance del DBMS. Son de uso exclusivo del DBA.

1. **Concepto de dependencia funcional. ¿Qué representa un determinante? ¿Es posible que exista una dependencia funcional establecida entre dos subconjuntos de atributos de la cabecera de una relación que puedan ser a la vez determinantes y dependientes? (Obviamente en dos dependencia funcionales distintas)**

Una dependencia funcional es un vínculo de muchos a uno que va de un subconjunto de atributos a otro dentro de una determinada relación.

Y depende funcionalmente de X

X determina funcionalmente a Y

X -> Y

Un determinante representa a un subconjunto de atributos, no vacio, que determina a otro subconjunto de atributos dentro de una relación. Sirven para poder identificar unívocamente a través de un subconjunto de atributos a otro. Especifican las dependencias entre atributos y como unos determinan a otros.

Si, es posible, por ejemplo en el caso de tener 2 claves candidatas ya que cualquier sea que elijas como pk final, ambas pueden determinar a la otra sin problemas.

1. **¿Qué características debe tener una relación en BCNF que posea anomalías de actualización? ¿Es posible que se deban a una dependencia funcional? Justifique.**

Para que una relación se encuentre en BCNF, la misma se debe encontrar en la 3FN y a su vez no debe poseer dependencias funcionales no triviales que no sean un conjunto de la clave candidata. La misma al poseer dependencias multivaluadas puede generar anomalías de actualización ya que tenemos atributos independientes entre sí dados por estas multivaluadas. No se deben a una dependencia funcional.

1. **Si en una relación R existe una Dependencia Funcional de tipo A -> B y A no es clave candidata de R, ¿Qué podemos decir que posee esta relación? ¿Por qué?**

Podemos decir que la relación no se encuentra en una forma normal óptima, ya que con la existencia de esta DF puede ni encontrarse en la segunda forma normal. Esto genera posibles problemas de actualización y posible redundancia de información.

1. **¿Qué son las reglas de inferencia? ¿Para qué sirven? Enumerarlas.**

* Reflexividad.
* Descomposición.
* Unión.
* Composición.
* Ampliación
* Transitividad.

**Anexo:**

* Definir “clave de indización”, “campo de agrupamiento”, “índice no denso” y “clave primaria”. ¿Cuales se hayan vinculadas entre sí? ¿De qué forma?
* Describa el funcionamiento de los arboles B y B+ por los índices. ¿Qué diferencia hay entre ellos? ¿Por que suele preferirse un árbol B+ como estructura de acceso a los archivos de datos?

Los arboles B y B+ poseen índices de múltiples niveles.

Los árboles b y b+ son estructuras de acceso de datos con forma de árbol que pueden ser usadas en bases de datos. La diferencia entre los árboles b y b+ es que el acceso en los arboles b+ es más rápido ya que todos los nodos de una misma altura se encuentran enlazados en forma de lista. Además los arboles B+ permiten lograr índices con menos niveles y más alta capacidad.

* El DA no es un usuario de la base de datos, ya que cumple otras funciones.
* ¿Que son los arboles de ocurrencia jerárquicos? ¿Cuál es su relación con los esquemas jerárquicos