

Archivos Secuenciales Indexados

Organización de Archivos

- 1.- Secuencial.
- 2.- Relativos.
Cálculo de Direcciones: Técnica Hash.
- 3.- Secuencial Indexado.

Secuencial Indexado

Archivos Indexados:

Son aquellos que tienen asociados un índice para mejorar la operación de búsqueda.

Características de un Índice:

1. Ocupa menos espacio que el archivo de datos y siempre se mantiene ordenado.
2. Cada entrada del índice contiene la clave y el apuntador al registro de datos.

Indices

- Un índice acelera la búsqueda en un archivo de datos. Permite acceso directo.
- Un índice contiene una colección de *data entries*.

Indices

- Cualquier subconjunto de atributos en una relación puede ser una clave de búsqueda.
- Las claves de búsqueda no son lo mismo que las claves primarias o alternas.

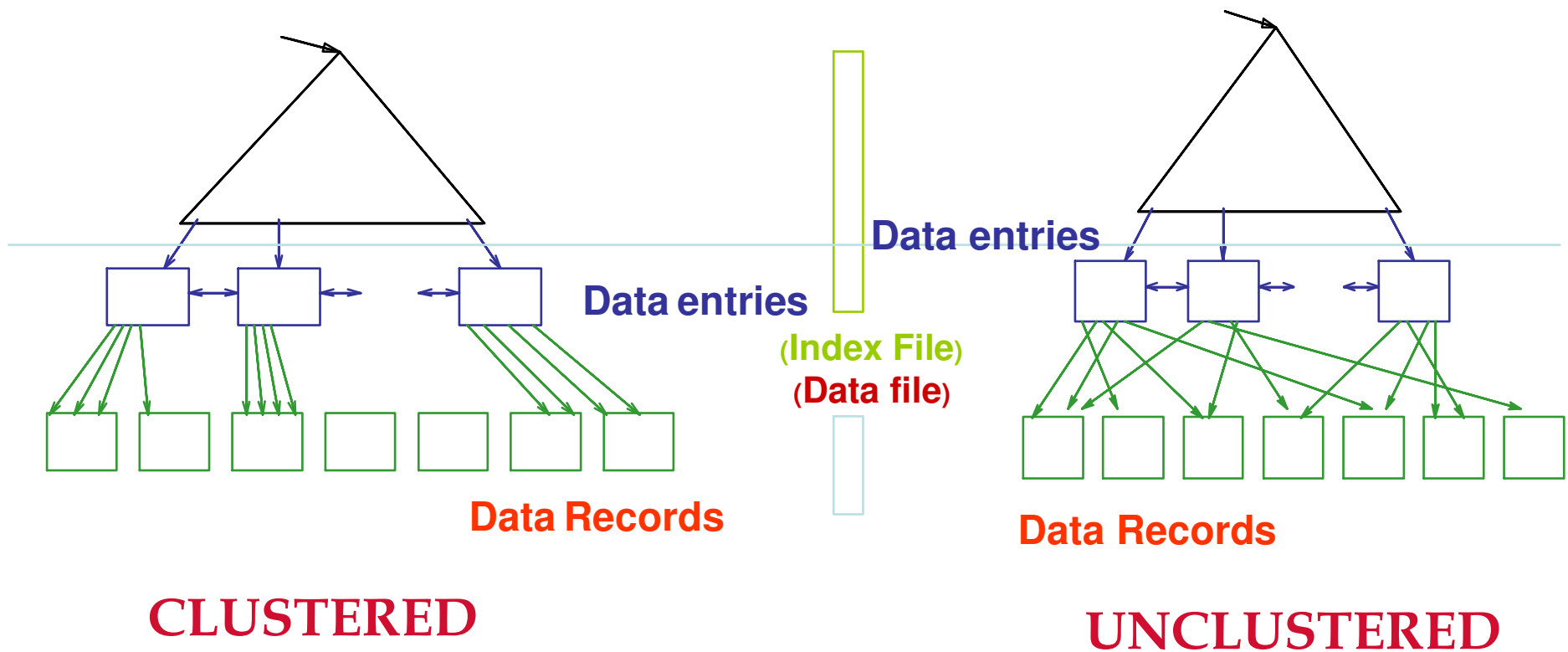
Data entries k^*

1. El data entry k^* es de hecho el registro de datos con clave de búsqueda k .
2. En el par (k, rid) rid es el identificador id de un registro de datos con clave de búsqueda k .
3. El data entry es un par $(k, \text{rid-list})$ donde rid-list es una lista de identificadores de registros de datos con clave k .

Clasificación de los Índices

- *Primarios vs. secundarios*: si la clave de búsqueda contiene una clave, entonces es primario.
- *Clustered vs. unclustered*: si el orden físico de los registros de datos es el mismo o cercano al orden lógico de los data entries, entonces se llama índice clustered.
- *Denso vs Disperso*: si por cada registro el índice tiene un par (*clave, dir*), es denso sino es disperso.

Propiedades de los Indices

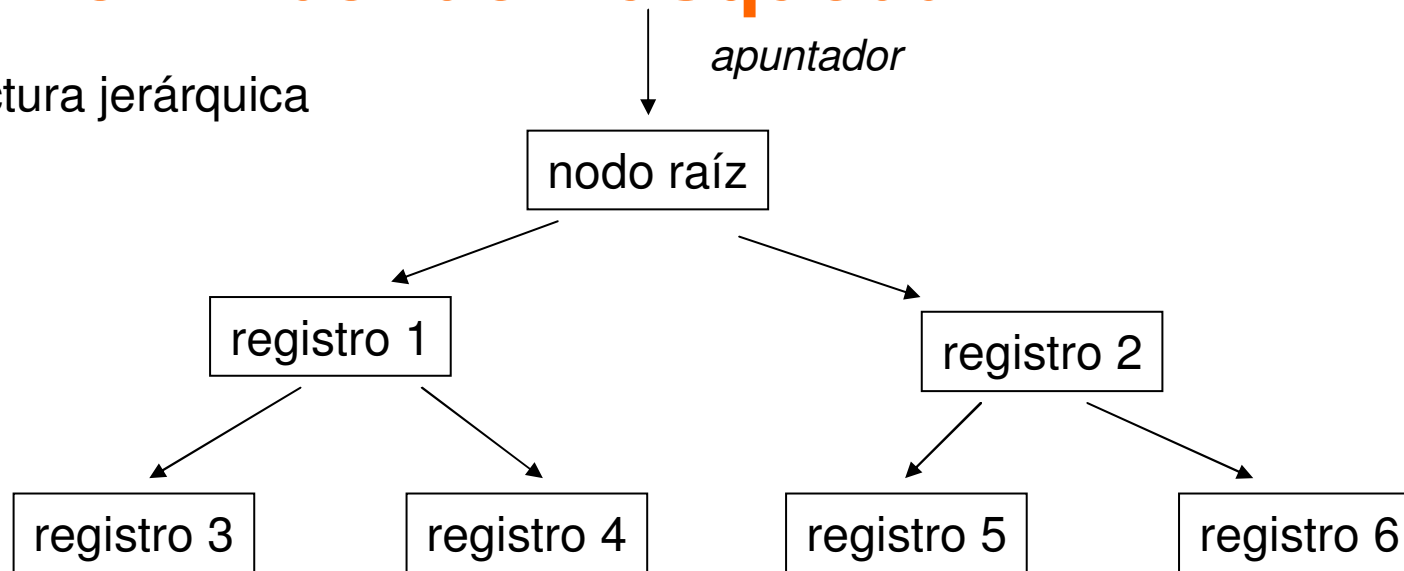


Archivo Secuencial Indexado

Una manera efectiva de organizar una colección de registros, cuando existe la necesidad tanto de acceder los registros secuencialmente, por algún valor de clave, como de obtenerlos individualmente, con esa misma clave, es la organización de archivos *secuenciales indexados*, éstos proporcionan la combinación de tipos de acceso que manejan un archivo secuencial y un archivo relativo. Dada su naturaleza son utilizados tanto para procesamiento por lote (*batch*) como interactivo.

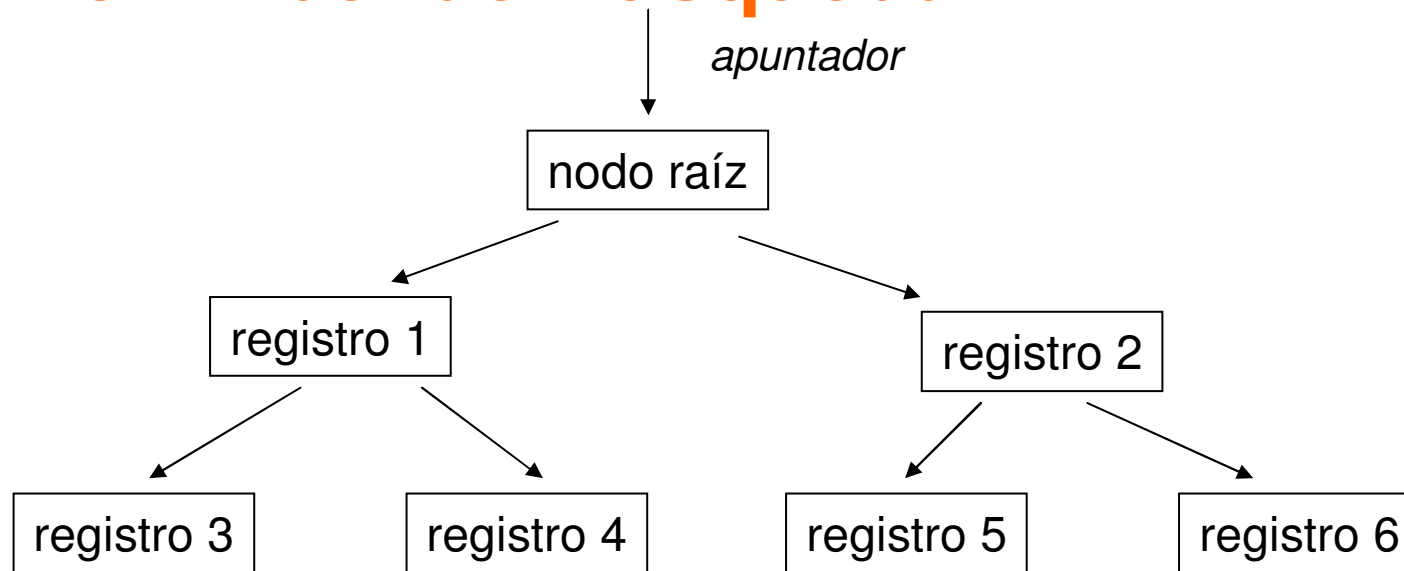
Previo: Árbol de Búsqueda

Estructura jerárquica



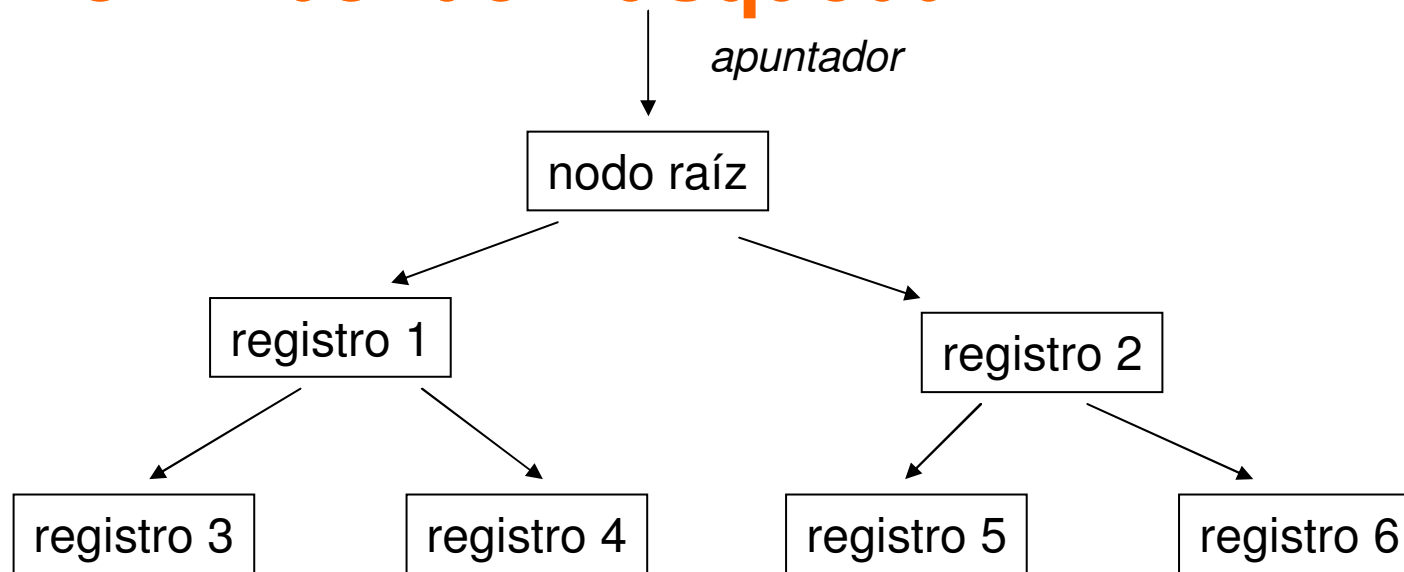
Se conectan todos los registros a través de los enlaces entre ellos.

Previo: Árbol de Búsqueda



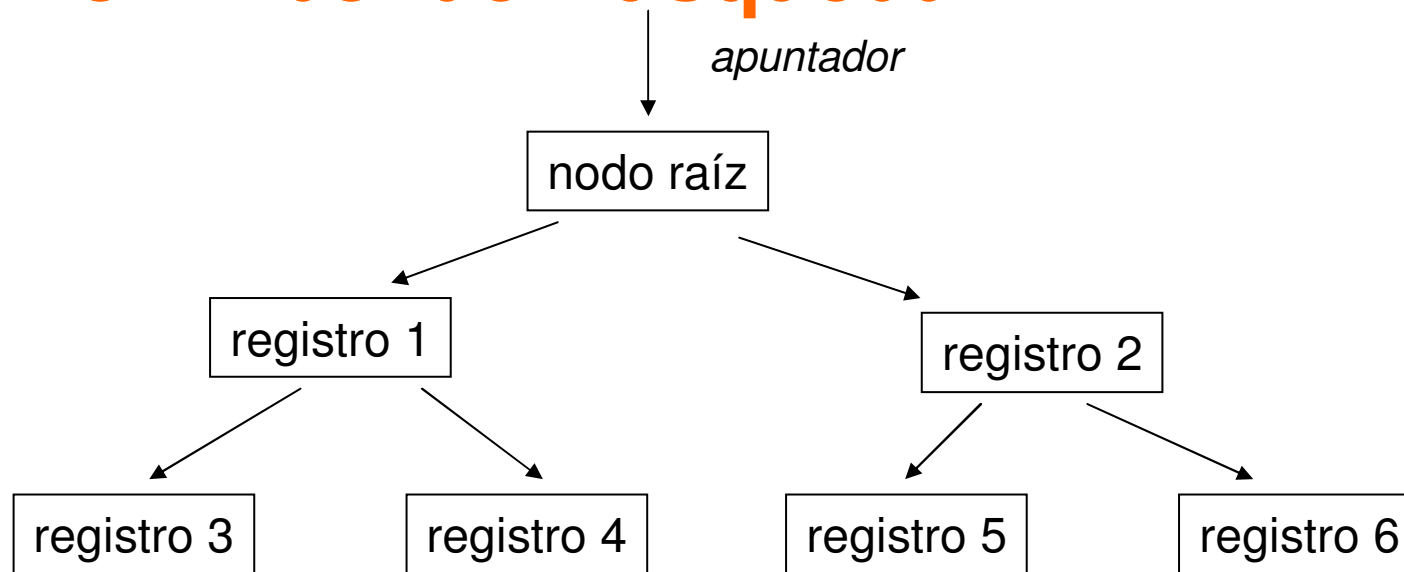
Todos los registros se distinguen por sus claves y en el árbol están ordenadas. Las claves del sub-árbol izquierdo son menores que las claves del sub-árbol derecho.

Previo: Árbol de Búsqueda



Para buscar el registro 6, sólo se hacen 3 preguntas en el árbol.
Si los registros estuvieran sólo en una estructura secuencial, se harían 6 preguntas hasta encontrar el registro 6.

Previo: Árbol de Búsqueda

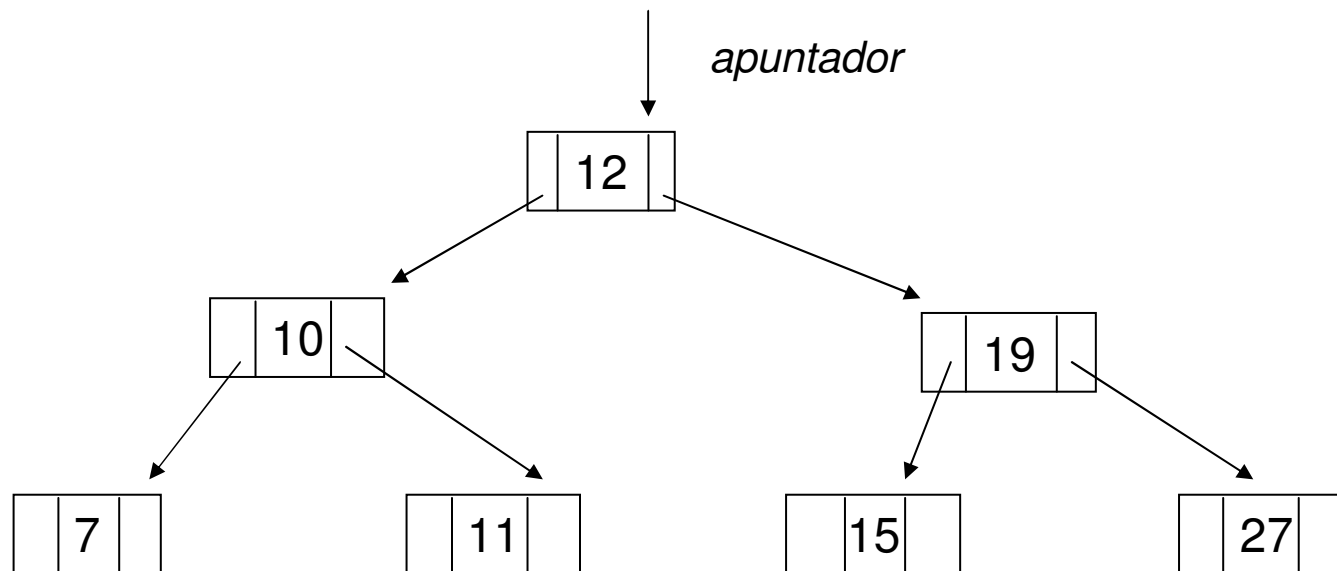


Los nodos en un árbol B contienen tres partes:

1. Un apuntador al sub-árbol izquierdo.
2. La información del registro.
3. Un apuntador al sub-árbol derecho.

Previo: Árbol de Búsqueda

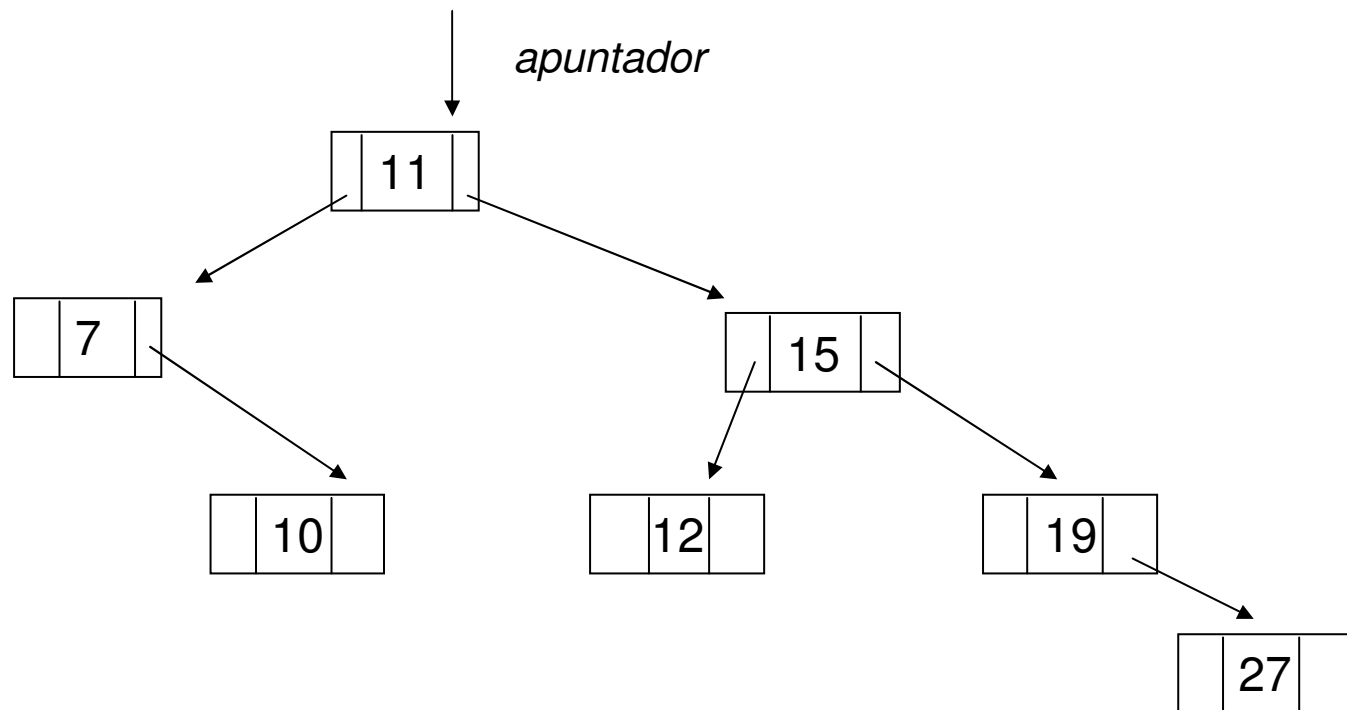
12, 19, 10, 15, 27, 11, 7



Orden del árbol: 2, porque cada raíz tiene a lo sumo 2 sub-árboles.
Se ingresan siempre empezando por la raíz

Previo: Árbol de Búsqueda

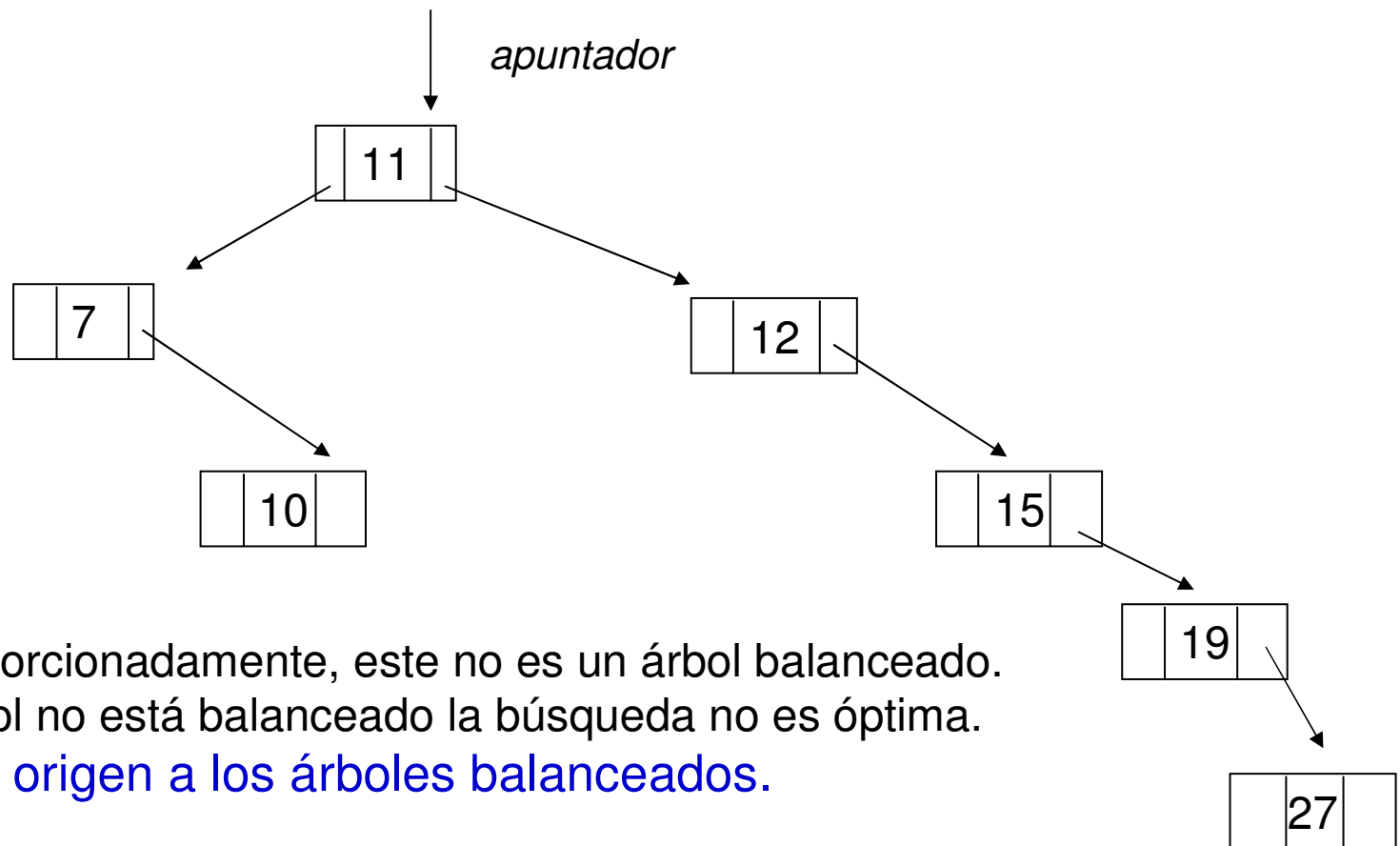
Si se cambia el orden de ingreso a: 11, 7, 10, 15, 12, 19, 27



Para llegar a 27 se hacen 4 preguntas a diferencia del árbol anterior en el cual se hacían 3 preguntas.

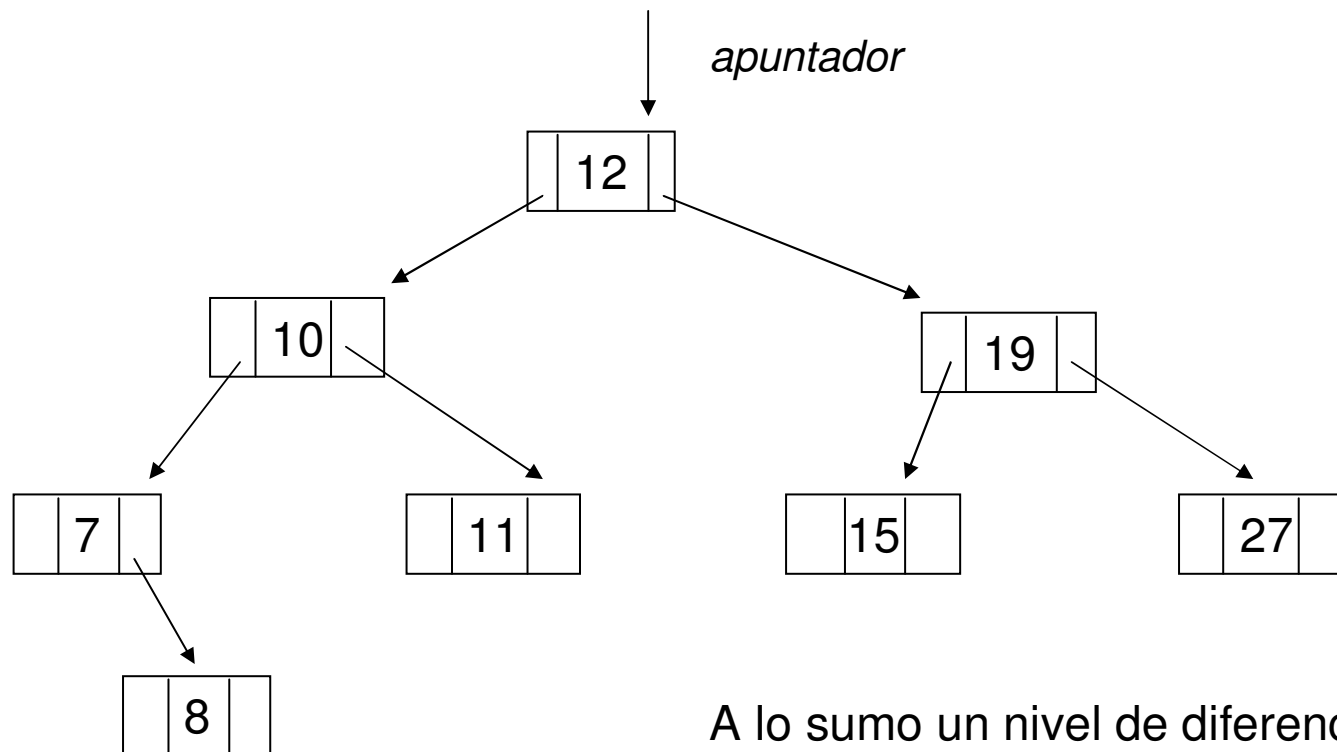
Previo: Árbol de Búsqueda

Si se cambia el orden de ingreso a: 11, 7, 10, 12, 15, 19, 27



Crece desproporcionadamente, este no es un árbol balanceado.
Cuando el árbol no está balanceado la búsqueda no es óptima.
Este ha dado origen a los árboles balanceados.

Previo: Árbol B

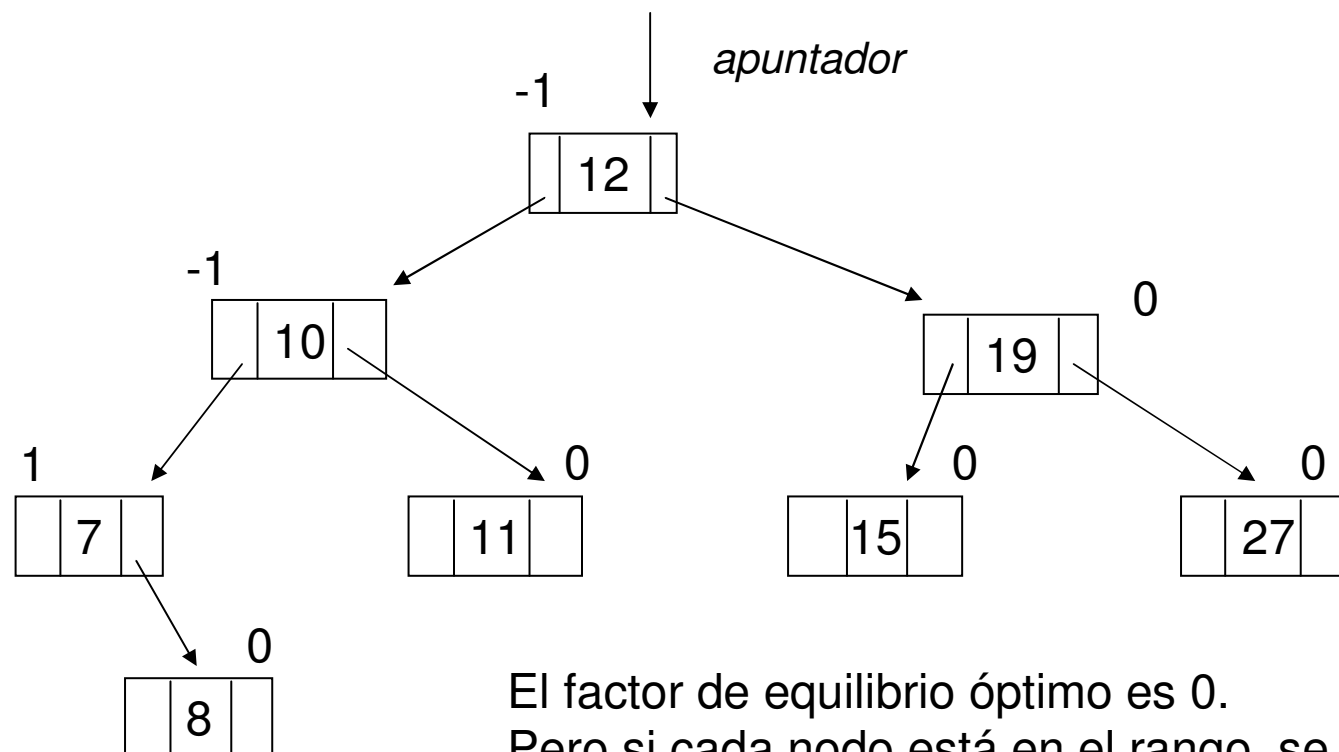


A lo sumo un nivel de diferencia entre los sub-árboles.

Se añade un elemento de información, llamado factor de equilibrio.

Previo: Árbol B

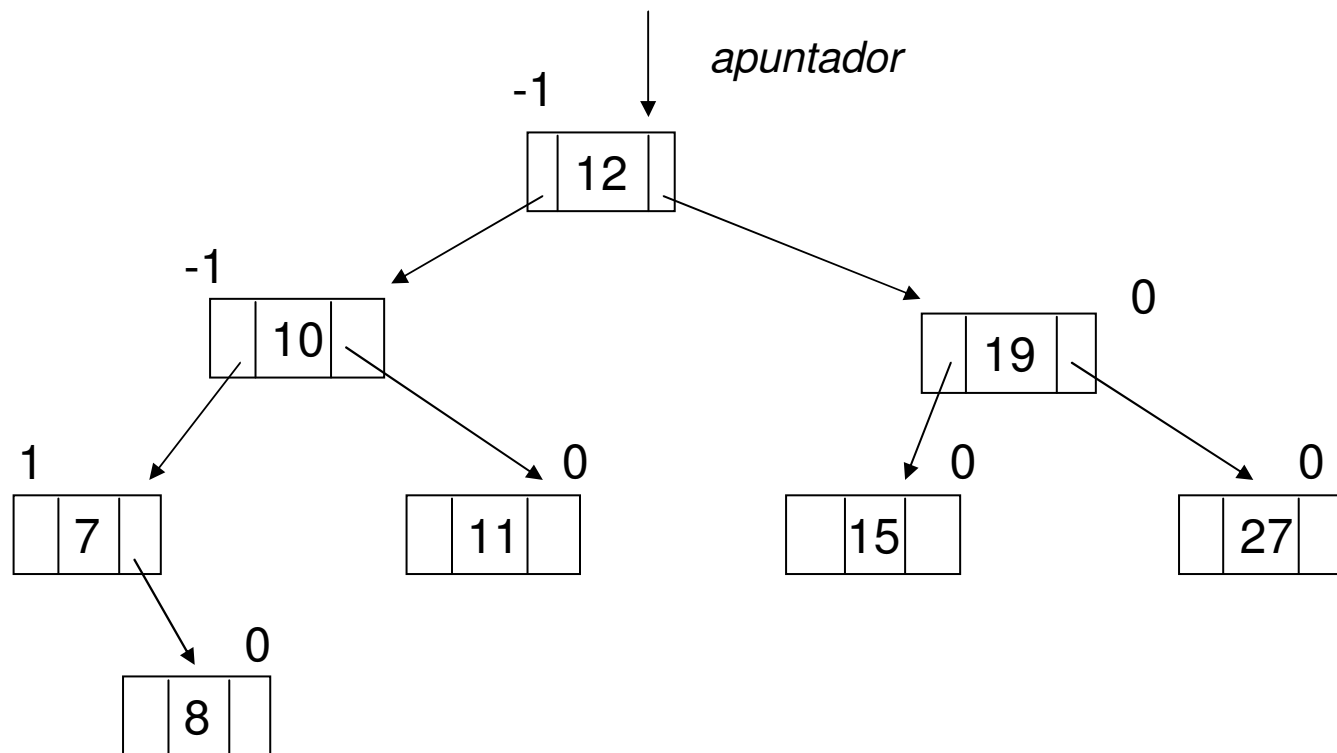
Valores del factor de equilibrio: -1, 0, 1



El factor de equilibrio óptimo es 0.
Pero si cada nodo está en el rango, se dice que el árbol está balanceado.
Si aumenta o disminuye, hay que balancearlo.

Previo: Árbol B – Desventajas.

- Mantener la estructura balanceada es sumamente costoso.
- Cada inserción o eliminación puede generar un proceso de balanceo.



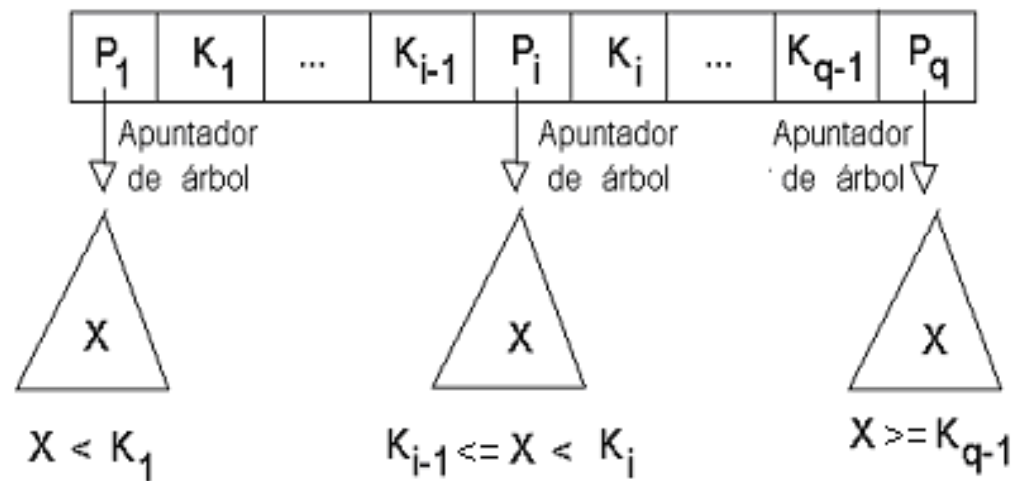
Estructura de Árbol B+

Una de las técnicas más usadas para la instrumentar la organización secuencial indexada de archivos es el uso de una variante de árbol B básico conocido como árbol B+.

Existen algunos métodos para estructurar las porciones de índices como datos secuenciales de un archivo secuencial indexado. El método más común es construir el índice como un árbol de valores claves.

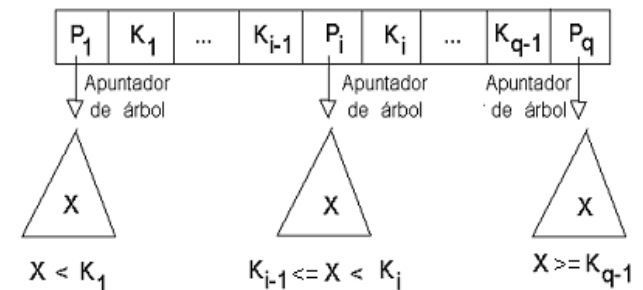
Estructura de Árbol B+.

Nodos Internos



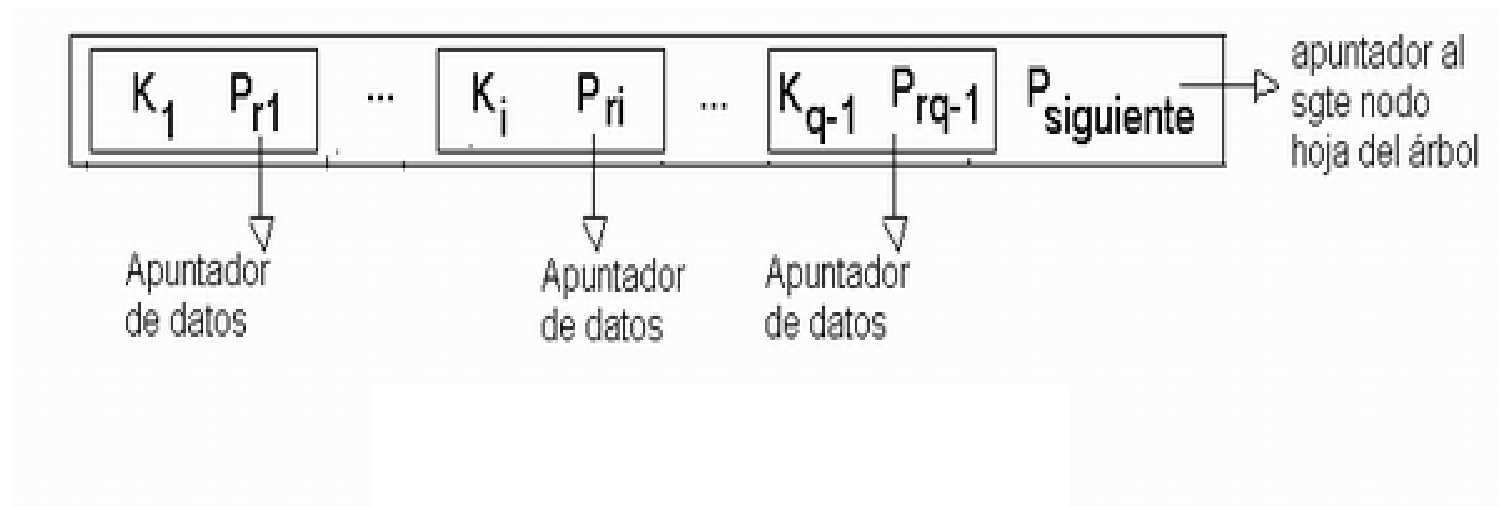
Estructura de Árbol B+. Nodos internos

1. Todo nodo interno es de la forma $\langle P_1, k_1, P_2, k_2, \dots, P_{q-1}, K_{q-1}, P_q \rangle$, donde $q \leq p$, cada P_i es un apuntador de árbol.
2. Dentro de cada nodo interno $K_1 < K_2 < \dots < K_{q-1}$
3. Para todos los valores X , del campo de búsqueda en el subárbol al cual apunta P_i , tenemos que $K_{i-1} \leq X < K_i$, para $1 < i < q$; $X < K_i$, para $i = 1$ y $X \geq K_{q-1}$ para $i = q$. En este caso pudiera encontrarse una X igual a K y debe seguirse hacia abajo, hasta llegar a un nodo hoja.
4. Cada nodo interno tiene, cuando más, p apuntadores de árbol.
5. Cada nodo interno, excepto la raíz y los nodos hoja, tienen por lo menos $p/2$ apuntadores de árbol. El nodo raíz tiene, como mínimo, dos punteros a nodos del árbol.
6. Un nodo interno con q apuntadores de árbol, $q \leq p$, tiene $q-1$ valores del campo de búsqueda



Estructura de Árbol B+.

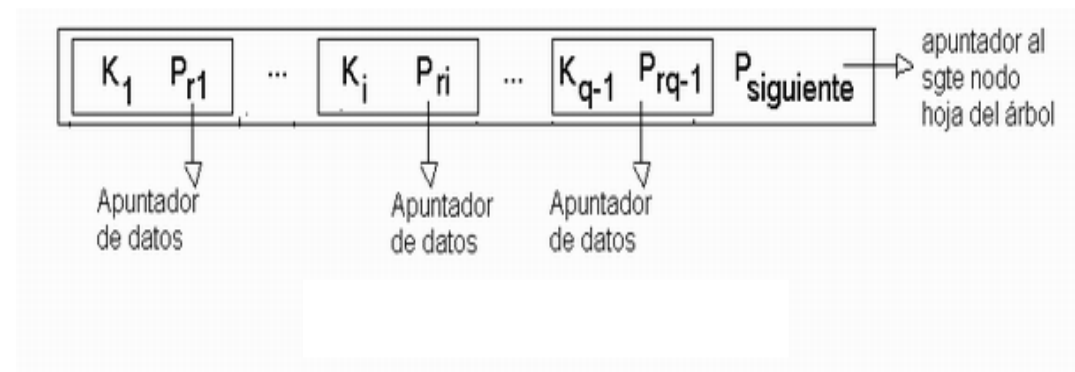
Nodos Hoja



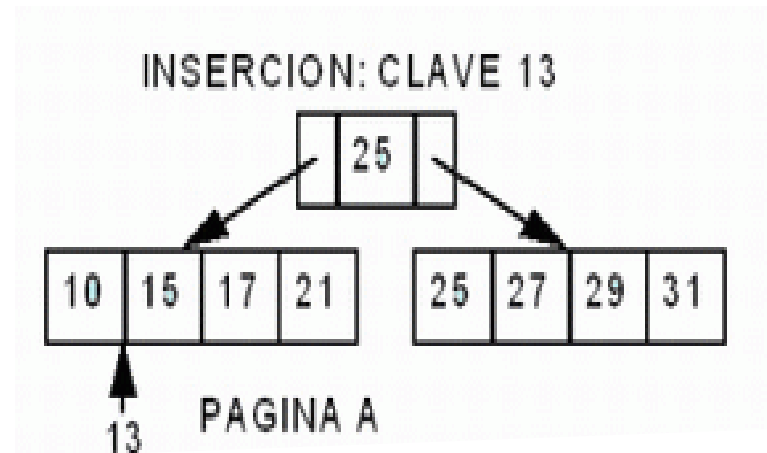
Estructura de Árbol B+. Nodos hoja

La estructura de los nodos hojas de un árbol B+ de orden p se define:

1. $\langle \langle k_1, Pr_1 \rangle, \langle k_2, Pr_2 \rangle, \dots, \langle k_{q-1}, Pr_{q-1} \rangle, P_{siguiente} \rangle$ donde $q \leq p$, cada Pr_i es un apuntador de datos y $P_{siguiente}$ apunta al siguiente nodo hoja del árbol B+.
2. Dentro de cada nodo hoja $K_1 < K_2 < \dots < K_{q-1}$, donde $q \leq p$
3. Cada nodo hoja tiene al menos $p/2$ valores
4. Todos los nodos hoja están en el mismo nivel



Estructura de Árbol B+. Inserción

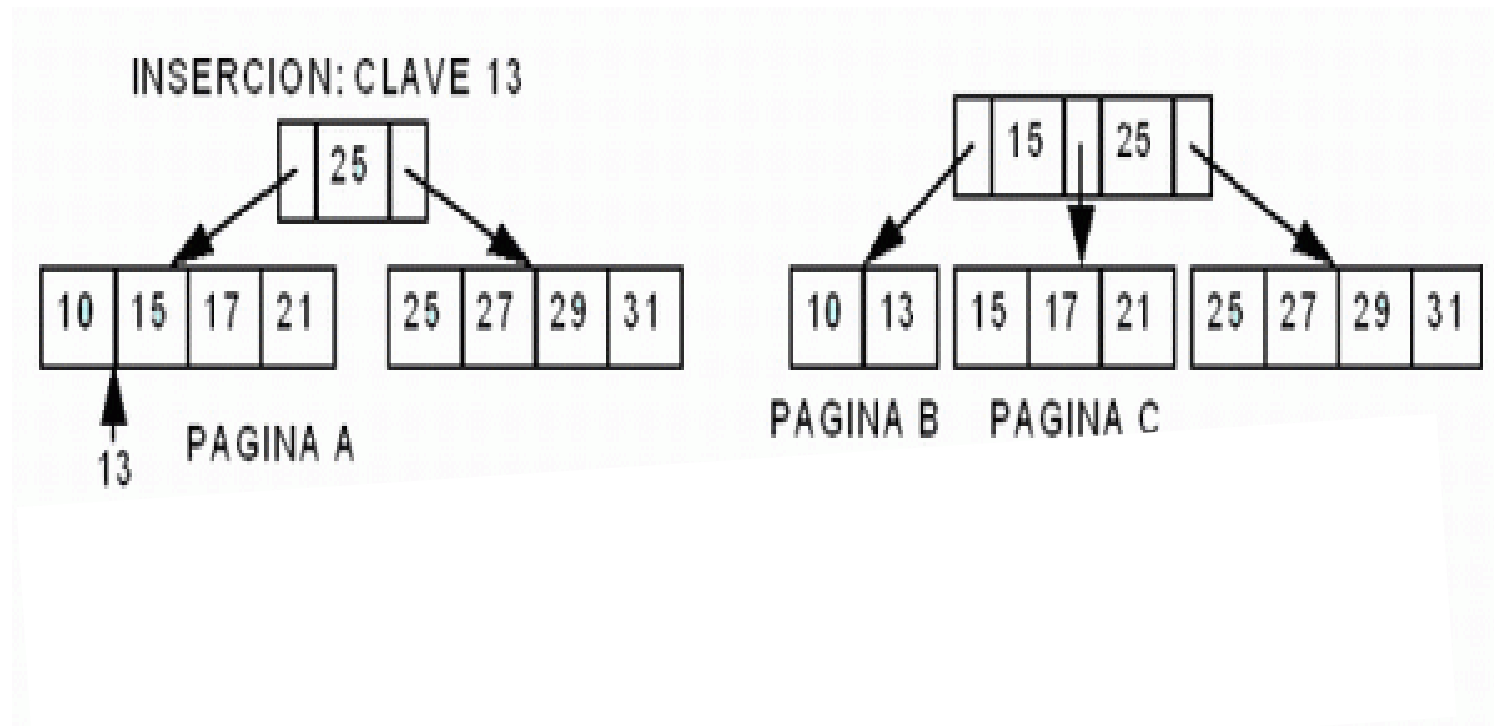


El proceso de inserción en árboles-B+ es relativamente simple, similar al proceso de inserción en árboles-B. La dificultad se presenta cuando desea insertarse una clave en un nodo que se encuentra lleno.

En este caso, el nodo afectado se divide en 2, distribuyéndose las claves de la siguiente forma: " las $p/2$ primeras claves en el nodo de la izquierda y las $p/2 + 1$ restantes claves en el nodo de la derecha".

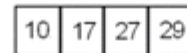
Una copia de la clave del medio sube al nodo padre.

Estructura de Árbol B+. Inserción

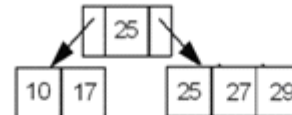


Estructura de Árbol B+. Inserción

a) INSERCIÓN: CLAVES 10, 27, 29 y 17



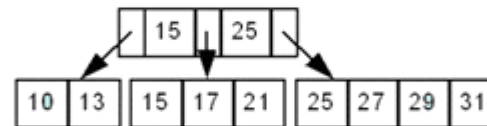
b) INSERCIÓN: CLAVE 25



c) INSERCIÓN: CLAVES 21, 15 y 31



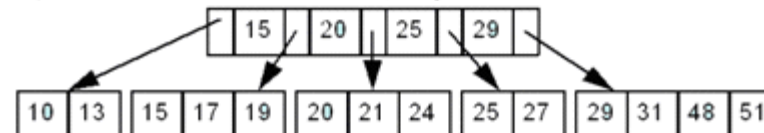
d) INSERCIÓN: CLAVE 13



e) INSERCIÓN: CLAVE 51



f) INSERCIÓN: CLAVES 20, 24, 48 y 19



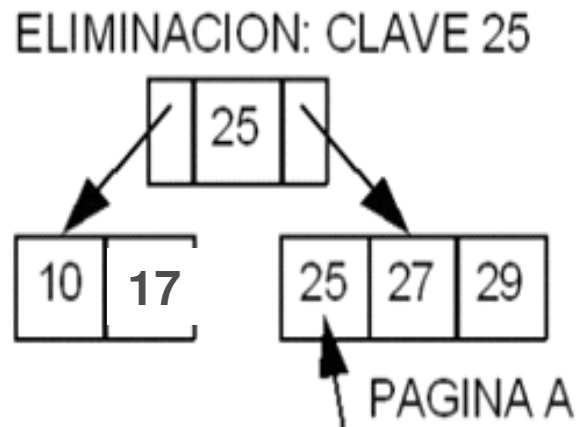
Estructura de Árbol B+. Eliminación

La operación de eliminación en árboles-B+ es más simple que en árboles-B.

Esto ocurre porque las claves a eliminar siempre se encuentran en las hojas. En general deben distinguirse los siguientes casos:

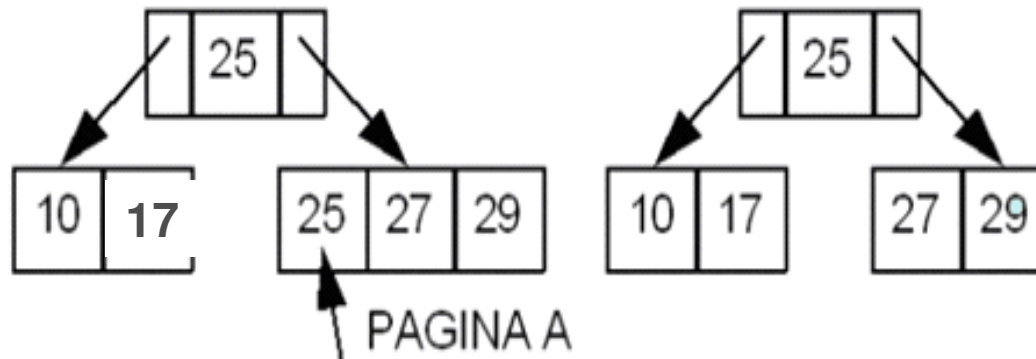
1. Si al eliminar una clave, la cantidad de claves queda mayor o igual que $p/2$ entonces termina la operación. Las claves de los nodos raíz o internos no se modifican por más que sean una copia de la clave eliminada en las hojas.
2. Si al eliminar una clave, la cantidad de claves queda menor que $p/2$ entonces debe realizarse una redistribución de claves, tanto en el índice como en las paginas hojas.

Estructura de Árbol B+. Eliminación



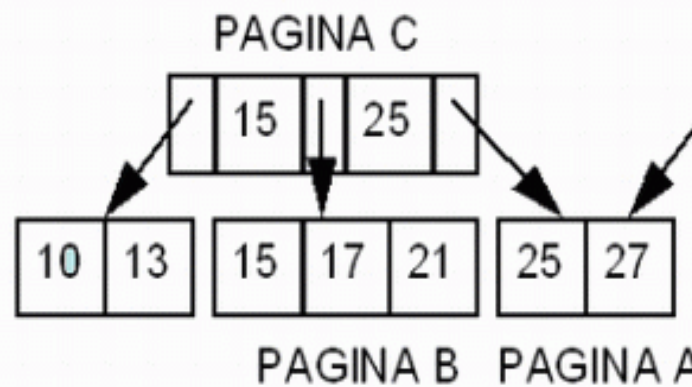
Estructura de Árbol B+. Eliminación

ELIMINACION: CLAVE 25

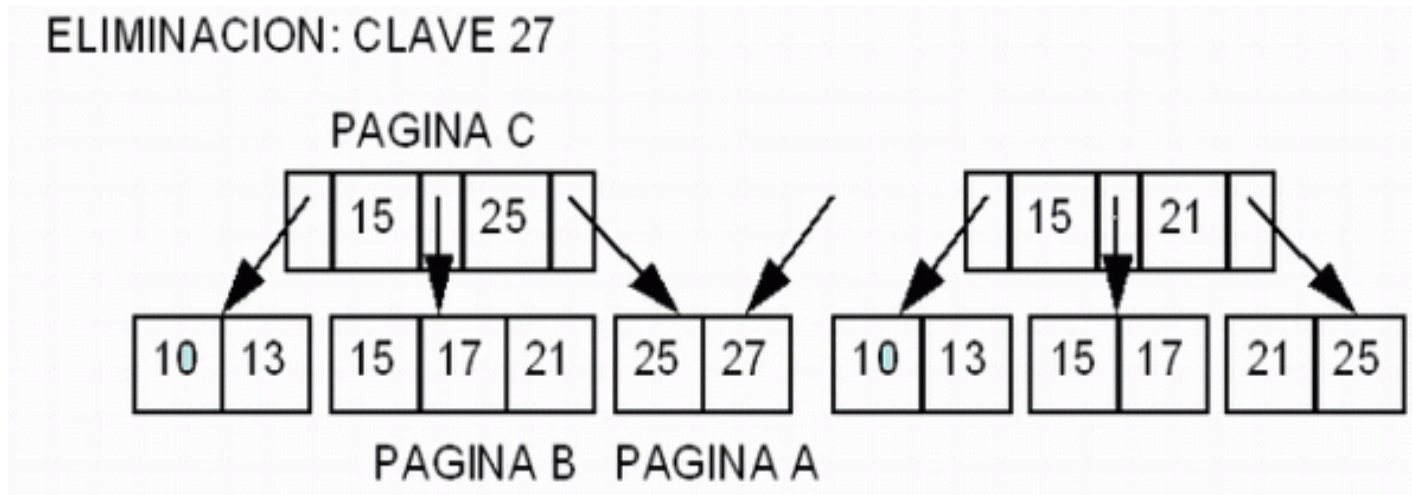


Estructura de Árbol B+. Eliminación

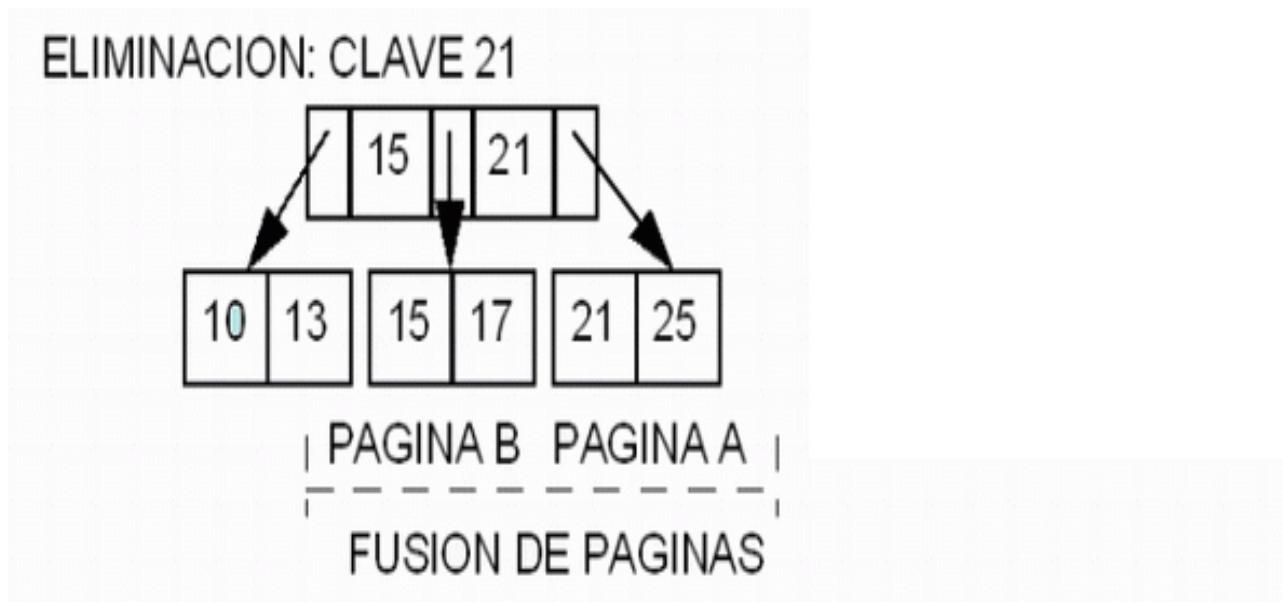
ELIMINACION: CLAVE 27



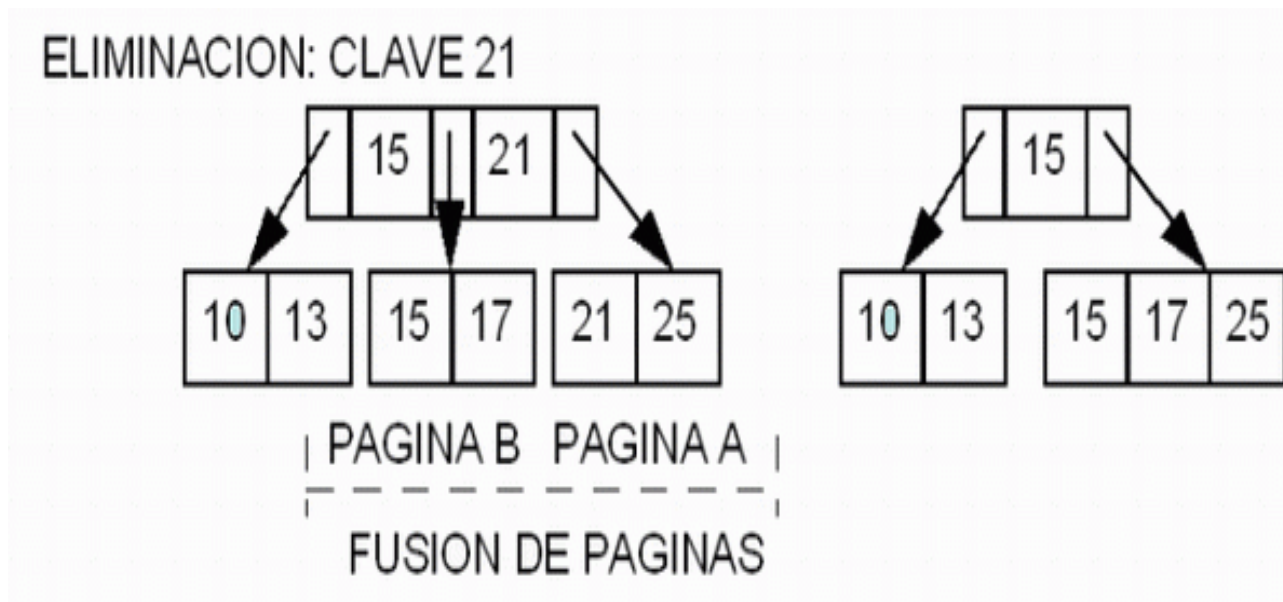
Estructura de Árbol B+. Eliminación



Estructura de Árbol B+. Eliminación



Estructura de Árbol B+. Eliminación



Fuentes consultadas:

[1] Ramakrishnan Raghu. ,
“Database Management Systems”.

[2] Prof. Elsa Liliana Tovar.
Notas de clase compiladas entre 1997-2004.

[3] Loomis M.
"Estructuras de Datos y Organización de Archivos"