

# Fundamentos de Organización de Datos

*Árboles B+*

# Árboles B+

- ✓ Constituyen una mejora sobre los árboles B, pues conservan la propiedad de acceso aleatorio rápido y permiten además un recorrido secuencial rápido.
- ✓ **Conjunto índice:** Proporciona acceso indizado a los registros. Todas las claves se encuentran en las hojas, duplicándose en la raíz y nodos interiores aquellas que resulten necesarias para definir los caminos de búsqueda.
- ✓ **Conjunto secuencia:** Contiene todos los registros del archivo. Las hojas se vinculan para facilitar el recorrido secuencial rápido. Cuando se lee en orden lógico, lista todos los registros por el orden de la clave.

# Búsqueda B+

La operación de búsqueda en árboles B+ es similar a la operación de búsqueda en árboles B. El proceso es simple, ya que todas las claves se encuentran en las hojas, deberá continuarse con la búsqueda hasta el último nivel del árbol.

# Inserción B+

Dificultad: Inserción en nodo lleno (overflow).

El nodo afectado se divide en 2, distribuyéndose las claves **lo más equitativamente posible**. Una **copia** de la clave del medio o de la menor de las claves mayores (casos de overflow con cantidad pares de elementos) se promociona al nodo padre. El nodo con overflow se divide a la mitad.



**La copia de la clave sólo se realiza en un overflow ocurrido a nivel de hoja.**

Caso contrario -> igual tratamiento que en árboles B.

# Bajas en B+

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 páginas hojas**. En general deben distinguirse los siguientes casos, dado un árbol B+ de orden  $M$ :

- Si al eliminar una clave, la cantidad de claves que queda es mayor o igual que  $\lceil M/2 \rceil - 1$ , 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.

# Bajas en B+

## Underflow

- Si al eliminar una clave, la cantidad de llaves es menor a  $\lceil M/2 \rceil - 1$ , entonces debe realizarse una **redistribución** de claves, tanto en el índice como en las páginas hojas.
- Si la redistribución no es posible, entonces debe realizarse una **fusión** entre los nodos.



# ***Políticas para la resolución de underflow:***

1. **Política izquierda:** se intenta redistribuir con el hermano adyacente izquierdo, si no es posible, se fusiona con hermano adyacente izquierdo.
2. **Política derecha:** se intenta redistribuir con el hermano adyacente derecho, si no es posible, se fusiona con hermano adyacente derecho.
3. **Política izquierda o derecha:** se intenta redistribuir con el hermano adyacente izquierdo, si no es posible, se intenta con el hermano adyacente derecho, si tampoco es posible, se fusiona con hermano adyacente izquierdo.
4. **Política derecha o izquierda:** se intenta redistribuir con el hermano adyacente derecho, si no es posible, se intenta con el hermano adyacente izquierdo, si tampoco es posible, se fusiona con hermano adyacente derecho.

# Árboles B+

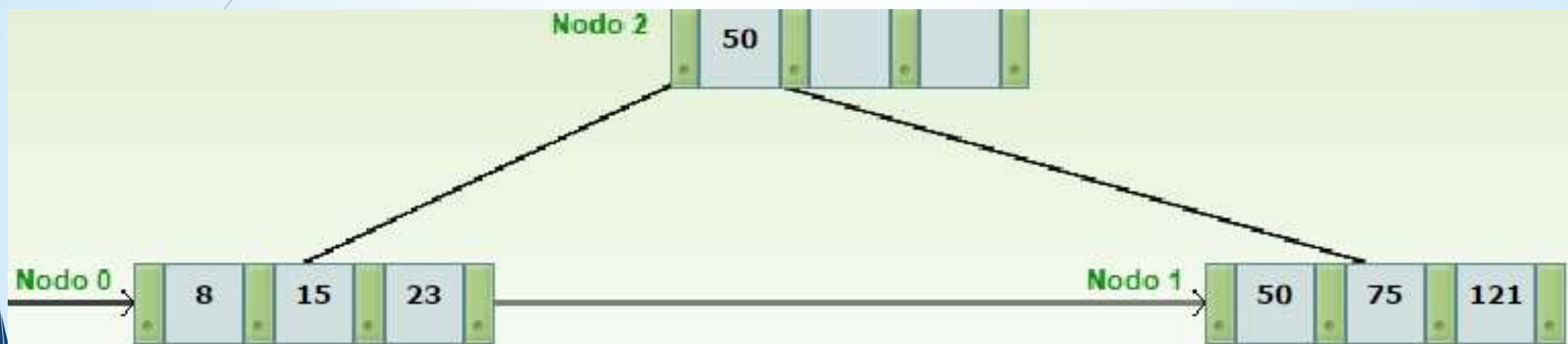
Ejemplo con árbol de orden 4

Claves:

+50, +75, +23



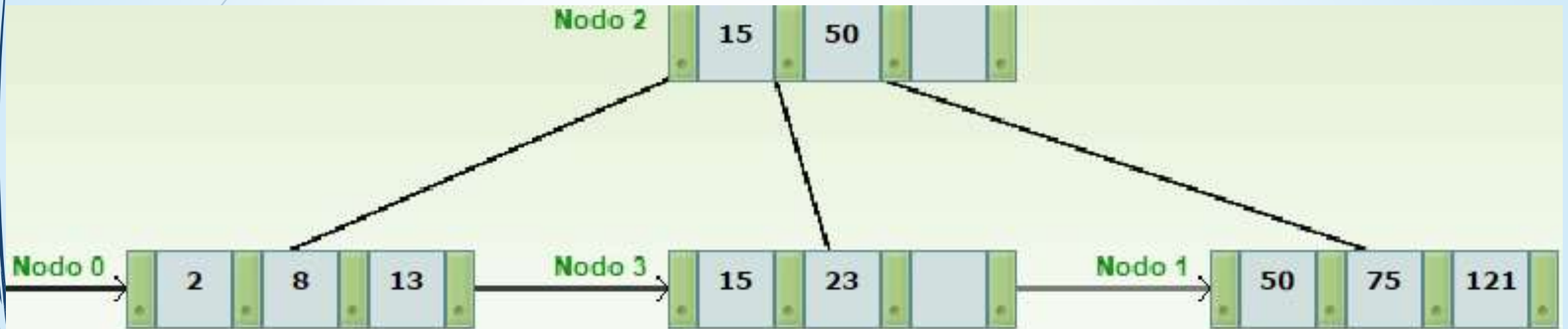
$\frac{8 \ 23}{\text{izq}}$      $\frac{50 \ 75}{\text{der}}$



+8 , +121, +15 , +2

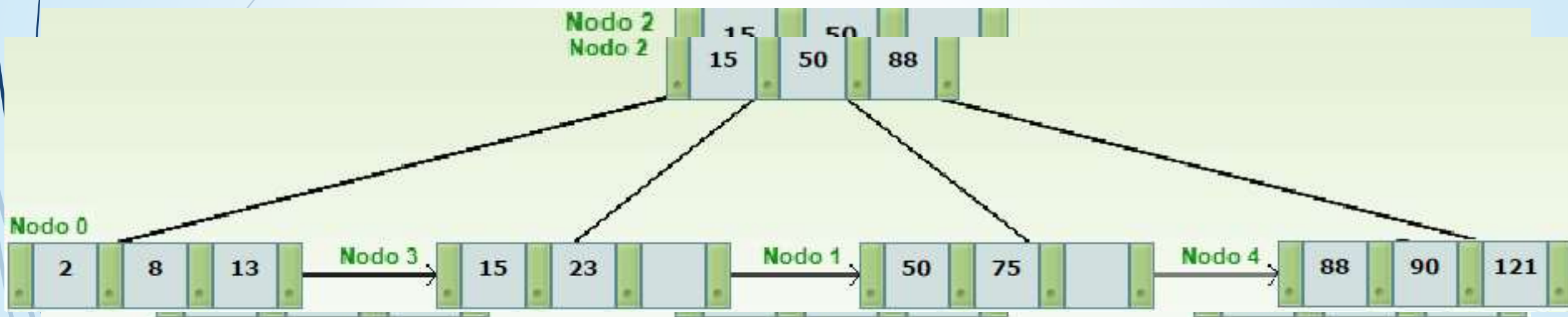
+2, +13, +88

$\frac{2 \ 8}{\text{izq}}$      $\frac{15 \ 23}{\text{der}}$



+88, +90, +100

50 75   88 121  
 izq   der

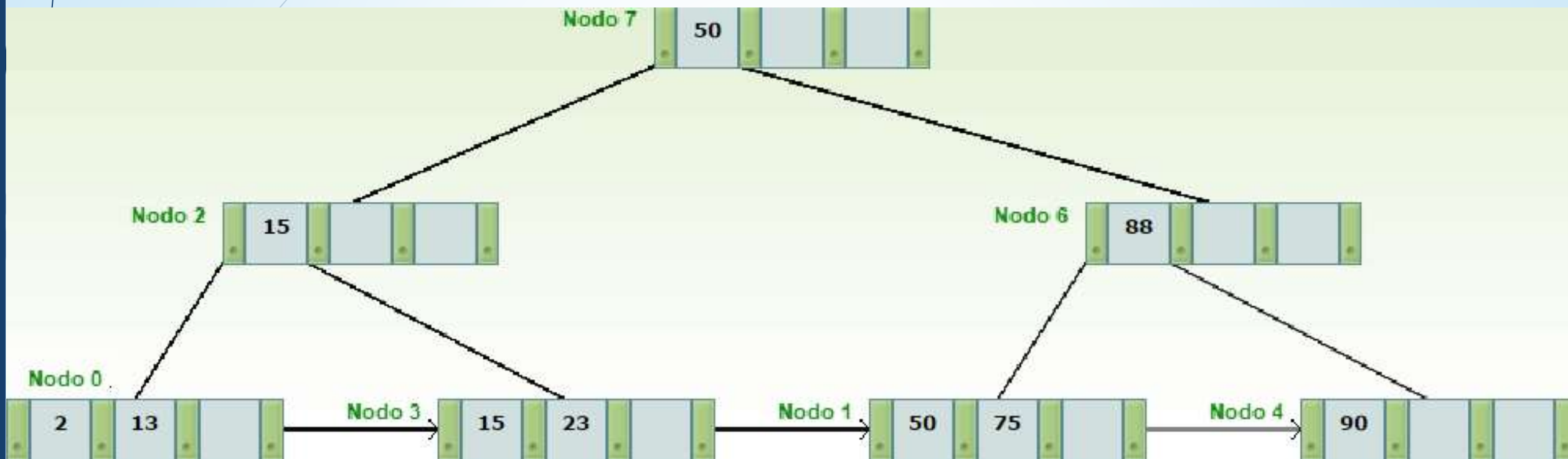


88 90   100 121  
 izq   der

15 50   88   100  
 izq   |   der  
 nueva raíz

Underflow en nodo 4. No es posible redistribuir y se fusionan los nodos 4 y 5. Se libera el nodo 5. Se propaga el underflow. Redistribución entre los nodos 2, 7 y 6.

15    50    88  
 —    |    —  
 izq   |   der  
 nueva raíz



+100, -8, -100, -121, -88

-90, -50

Underflow en nodo 1. No es posible redistribuir y se fusionan los nodos 1 y 4. Se libera el nodo 4. Se propaga el underflow. No es posible redistribuir, se fusiona nodo 2 y 6 liberando el nodo 6 y decrementando la altura del árbol liberando el nodo 7.

