

## Explicación 4 – Árboles B

¿Qué es un underflow?

**Respuesta:** cuando quedan menos claves de lo mínimo permitido en un árbol determinado. El mínimo lo establece el orden  $n$ , mediante la función:  $min = n/2 - 1$

Overflow: cuando se supera la cantidad máxima de claves en un nodo

Supóngase un árbol con capacidad para 3 claves, que tiene los siguientes nodos:

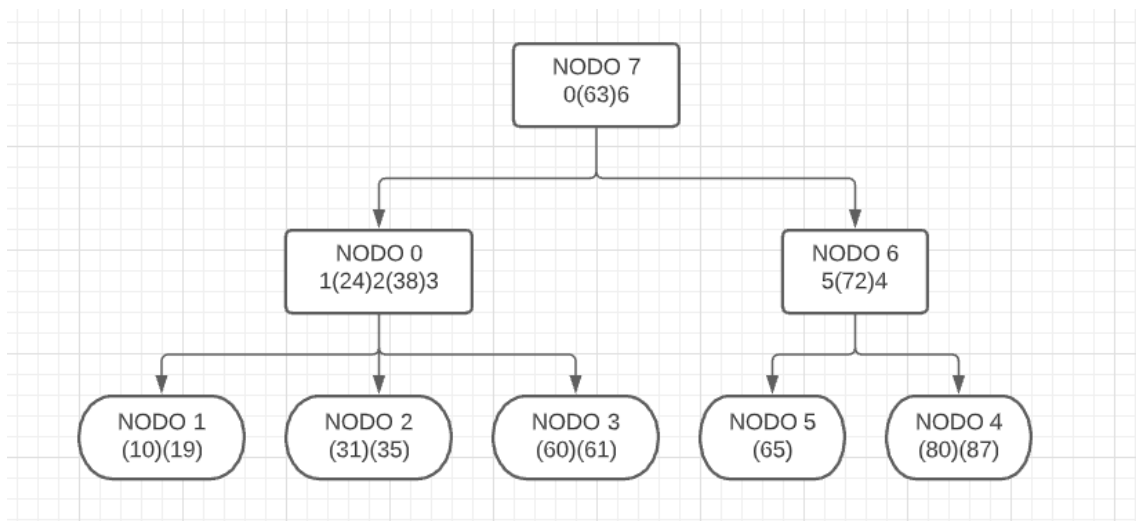
- Nodo 0: 1(24)2(38)3(72)4
- Nodo 1: (10)(19) hoja
- Nodo 2: (31)(35) hoja
- Nodo 3: (60)(63)(65) hoja
- Nodo 4: (80)(87) hoja

**No chequeado:** el nodo 0 es la cabecera, el número entre paréntesis (clave) indica cuál es valor máximo de clave que podés encontrar yendo por el nodo apuntado.

Se pide realizar el alta de la clave 61.

**Resolución:** según el nodo 0, tengo que ir al nodo 3 (porque 61 está entre el 38 y 72)

- Llego al nodo 3, quiero insertar pero no tengo espacio (overflow)
- Debo dividir las claves, el nodo 3 con (60)(61) y un nuevo nodo 5 con (65)
- ¿Qué paso con (63)? La menor de las mayores del nuevo debe propagarse al padre.
- Oh sorpresa, no hay espacio en el nodo 0. Hay que volver a dividir las claves
- El nodo 0 queda con (24)(38), un nuevo nodo 6 con (72) y una nueva raíz 7 con (63)
- En este procedimiento es importante ir actualizando los punteros.



El costo de la operación: L0, L3, E3, E5, E0, E6, E7 (L = leer, E = escribir)

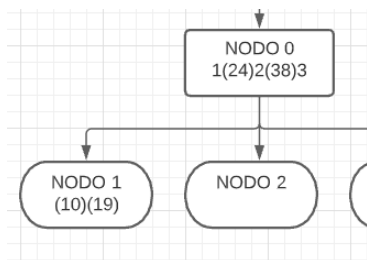
## Capítulo 1 – Bajas y Underflow

Cuando se da de baja una clave de un nodo existe el riesgo de que dicho nodo quede en underflow, es decir, con menos claves de lo mínimo permitido. Por ejemplo, en un árbol de orden 4, no puede haber nodos que tengan menos de 1 clave (no se permiten vacíos).

Supongamos que tenemos el árbol resultante de la página anterior, de orden 4, y si desea eliminar la clave 31. Esto requerirá leer el nodo 7 (raíz), luego el 0 (menores a 63) y por último el nodo 2 (entre 24 y 38). Al borrar el 31, el nodo 2 se queda con una única clave (permitido).

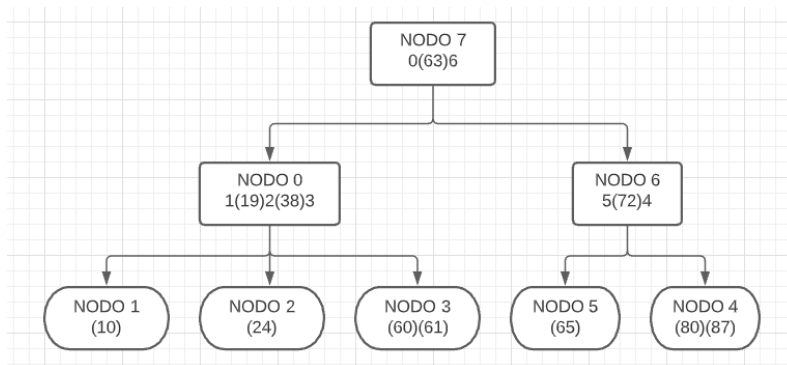
### Sección 1.1 – Política izquierda

Ahora se desea eliminar la clave 35. Ídem anterior, pero el nodo 2 se queda sin claves (underflow) así que usaremos una política (por ejemplo, izquierda) para resolver este problema.



El hermano adyacente de 2 es 1, él tiene 2 claves. Se le puede extraer una clave sin producir un nuevo underflow. La operación de re-distribución involucra también al nodo padre (0).

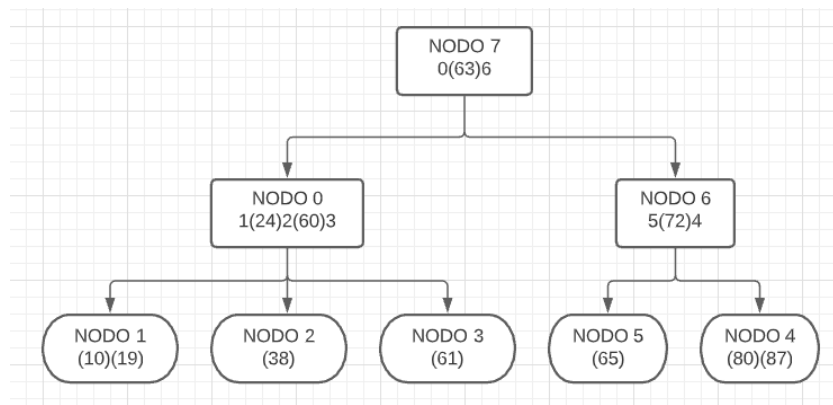
La clave 10 se queda en el nodo 1. Las claves 19 y 24 son tentativas de quedar en el nodo 2, pero la menor de ellas se queda en el nodo padre. Entonces la clave 19 va al nodo 0, y la clave 24 va al nodo 2, de modo de no romper el ordenamiento.



El costo resultante es entonces: L7, L0, L2, L1, E1, E2, E0 (izq-der-padre)

### Sección 1.2 – Política derecha

Supongamos la misma situación que en 1.1 (baja de la clave 35). Para resolver el underflow, se consulta al nodo 3, que en este caso tiene 2 claves, suficientes para la re-distribución. La menor de ellas (60) debe pasar al nodo 0 (padre), y la clave 38 del padre debe pasar al nodo 2.



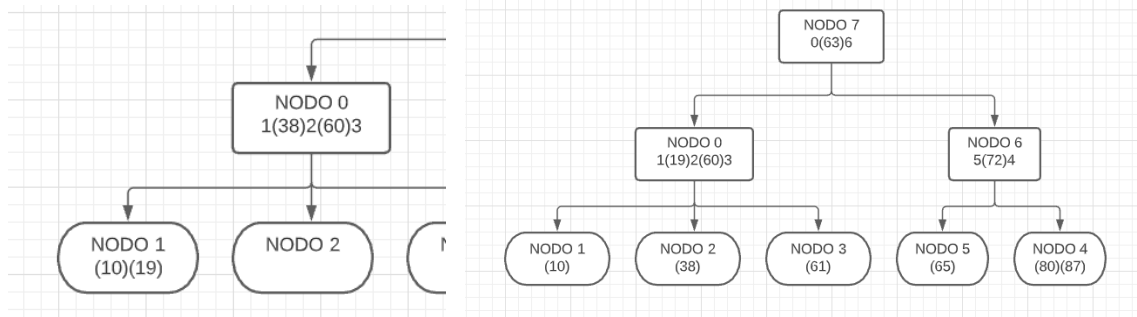
El costo de la operación resulta: L7, L0, L2, L3, E2, E3, E0 (izq-der-padre)

### Sección 1.3 – Baja en nodo interno, política izquierda

Supongamos el último árbol resultante de la página anterior. Se desea dar de baja la clave 24.

Primero se debe encontrar dicha clave. Leemos el nodo 7 (raíz), y luego el 0 (menores a 63), que es justamente donde está el 24. Debido a que el nodo 0 no es una hoja, se busca el menor del subárbol derecho (el apuntador luego de 24 dice que busquemos por el nodo 2, en este caso es una hoja con una única clave, así que tomamos su clave 38, aunque se producirá underflow).

Desde este punto operamos de igual manera que en 1.1, moviendo la clave 19 del hermano adyacente izquierdo al nodo 0 (padre), y la clave 38 del padre hacia el nodo 2.

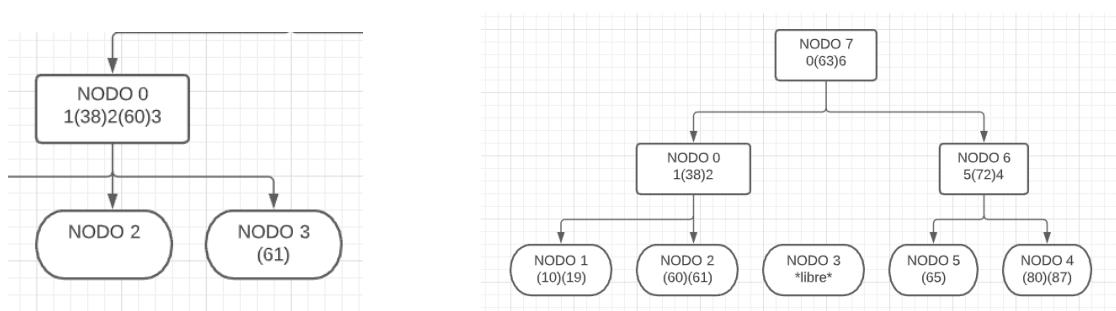


El costo de la operación en este caso resulta: L7, L0, L2, L1, E1, E2, E0 (izq-der-padre)

**Notar que:** la política de izquierda solo se utiliza cuando se produce el underflow (nodo 2).

### Sección 1.4 – Baja en nodo interno, política derecha (plus fusión)

Ídem anterior, pero cambia la re-distribución para resolver el nodo 2. En este caso, se consulta al hermano adyacente derecho (nodo 3), que tiene una única clave (61), por lo que resultará en underflow si se la extraemos. Entonces, en lugar de re-distribuir, se procede a ✨fusionar✨



El costo de la operación resulta: L7, L0, L2, L3, E2, E0 (izq-der-padre)

**Advertencia:** cuando se libera un nodo, NO se debe escribir dicha operación como escritura.

**Nota:** si se debe realizar una futura inserción, debe usarse un nodo libre como 1° opción.

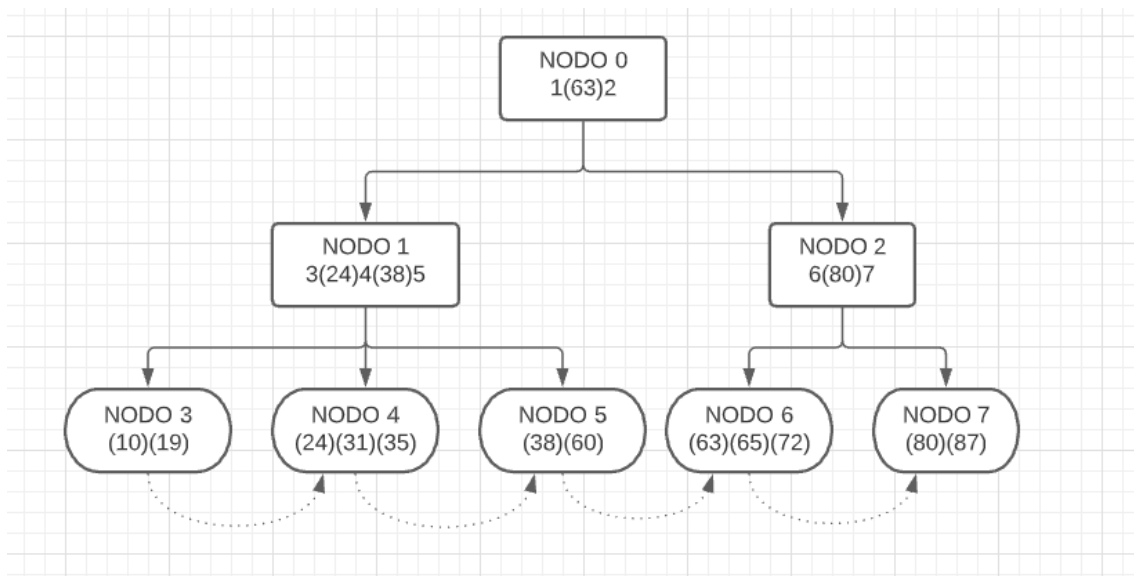
## Capítulo 2 – Árboles B+

Solo contienen claves en las hojas. Los nodos internos únicamente tienen separadores.

Además, todas las hojas están enlazadas, y la última apunta a un nodo inválido (-1).

### Sección 2.1 – Altas

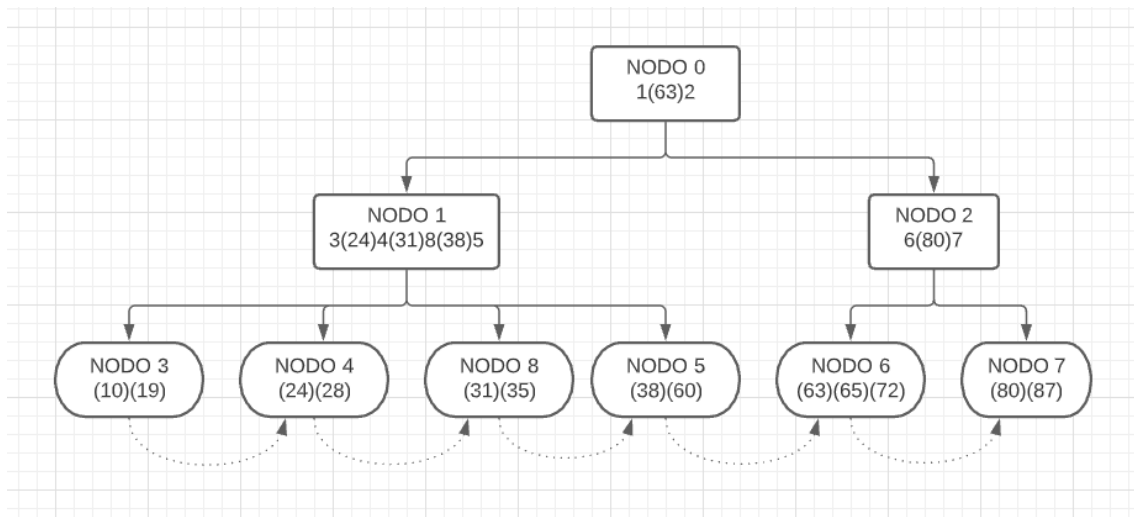
Supongamos el árbol de la página siguiente, con capacidad para 3 claves.



**Objetivo:** dar de alta la clave 28

**Resolución:** se lee el nodo 0 (raíz), luego el 1 (menores a 63) y luego el 4 (entre 24 y 38). No puedo insertar directamente porque el nodo 4 está completo (se produce overflow). Se procede a dividir, quedando las claves 24 y 28 en el nodo 4, y las claves 31 y 35 en un nuevo nodo 8.

El nodo 4 debe apuntar al nodo 8 (enlazar), y luego, el nodo 8 debe apuntar al nodo anteriormente referenciado por el nodo 4, es decir, el nodo 8 apunta al nodo 5. Además, la **llave** 31 se agrega al nodo 1 (padre) para que los separadores sigan siendo válidos, y por suerte tiene espacio.



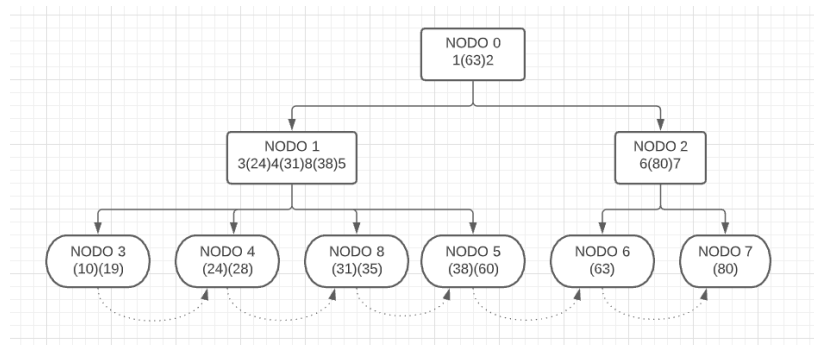
El costo de la operación resulta: L0, L1, L4, E4, E8, E1

**Posible error en la filmina:** aparece duplicado L4 en el costo, y no se menciona E4

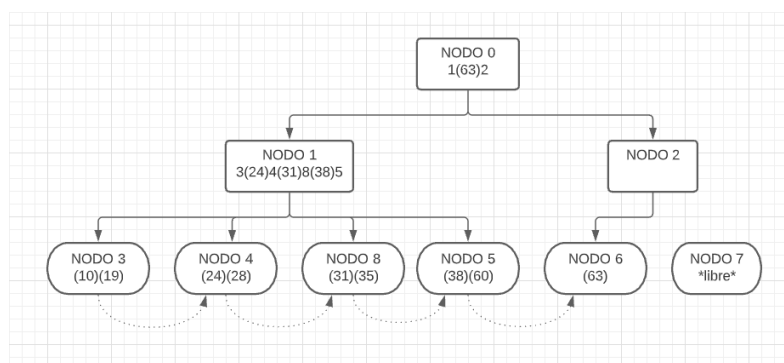
## Sección 2.2 – Bajas usando política izquierda y derecha

Al igual que en árboles B, también existen políticas para resolver underflow.

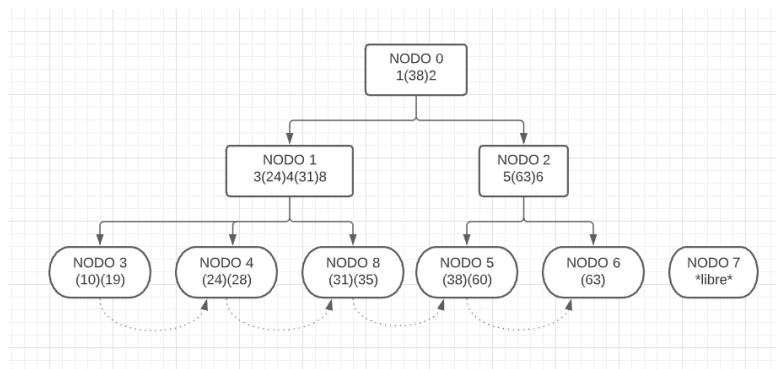
**Objetivo:** dar de baja la clave 80 del árbol de la sig pág, usando política izquierda y derecha.



**Resolución:** se lee el nodo 0 (raíz), luego el 2 (desde 63) y luego el 7 (desde 80). Al borrar queda vacío, así que tratamos de re-distribuir con el hermano adyacente izquierdo, pero no es posible porque **está al mínimo**, entonces intento re-distribuir con el hermano adyacente derecho, pero, oh sorpresa, no existe. ¿Qué se hace en este escenario? **Fusionar** los nodos 6 y 7.



¿Esto está bien? **NO**, porque el nodo 2 (padre) quedó en underflow. De manera recursiva, se utiliza la política (no literalmente) para re-distribuir hasta que resolver todos los problemas. Ahora se consulta al nodo 1 (hermano izquierdo del 2), que por suerte no está al mínimo, así que re-distribuimos, recordando también que el nodo 0 está involucrado. Específicamente, se traslada la clave 38 al nodo 0 (padre), y se mueve la clave 63 al nodo 2. Este es el resultado final:



**Notar que:** en los árboles B+, debido a que no hay claves en los nodos internos, la hoja involucrada puede cambiar de padre (es el caso del nodo 5, que ahora es hijo del nodo 2). Esto simplemente se realiza trasladando el separador entre nodos internos hermanos.

El costo de la operación en este caso es: L0, L2, L7, L6, L1, E6, E1, E2, E0

**Recordatorio:** liberar un nodo no es una escritura, así como tampoco el cambio de padre.

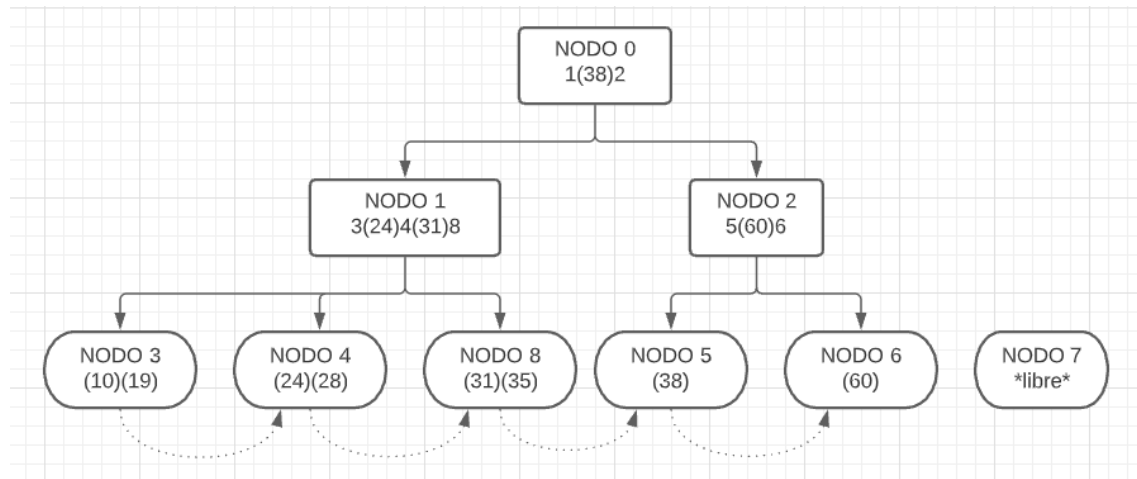
**Error en la filmina:** está duplicado L1 en el costo, pero queda claro que es un error de tipeo.

Hagamos una última operación: fulminemos la clave 63.

Y sucede algo que seguro no te lo esperabas.... Un underflow (ah s\*\*\* here we go again)

La parte buena es que la clave 63 está en el nodo 6, y su nuevo hermano no está al mínimo.

Prosigo a re-distribuir, recordando que en el nodo 2 (padre) no hay claves, sino separadores.



¿Esto es correcto? **Sí**, todos los nodos tienen al menos una clave, que es lo mínimo permitido.

Costo de la operación en este caso: L0, L2, L6, L5, E5, E6, E2 (dictado en el video)

Para pensar: ¿cuál es el costo de dar de baja la clave 24? ¿y si luego elimino la 28?