

OpenCourseWare
Grado Ingeniería Informática
Estructura de Datos y Algoritmos

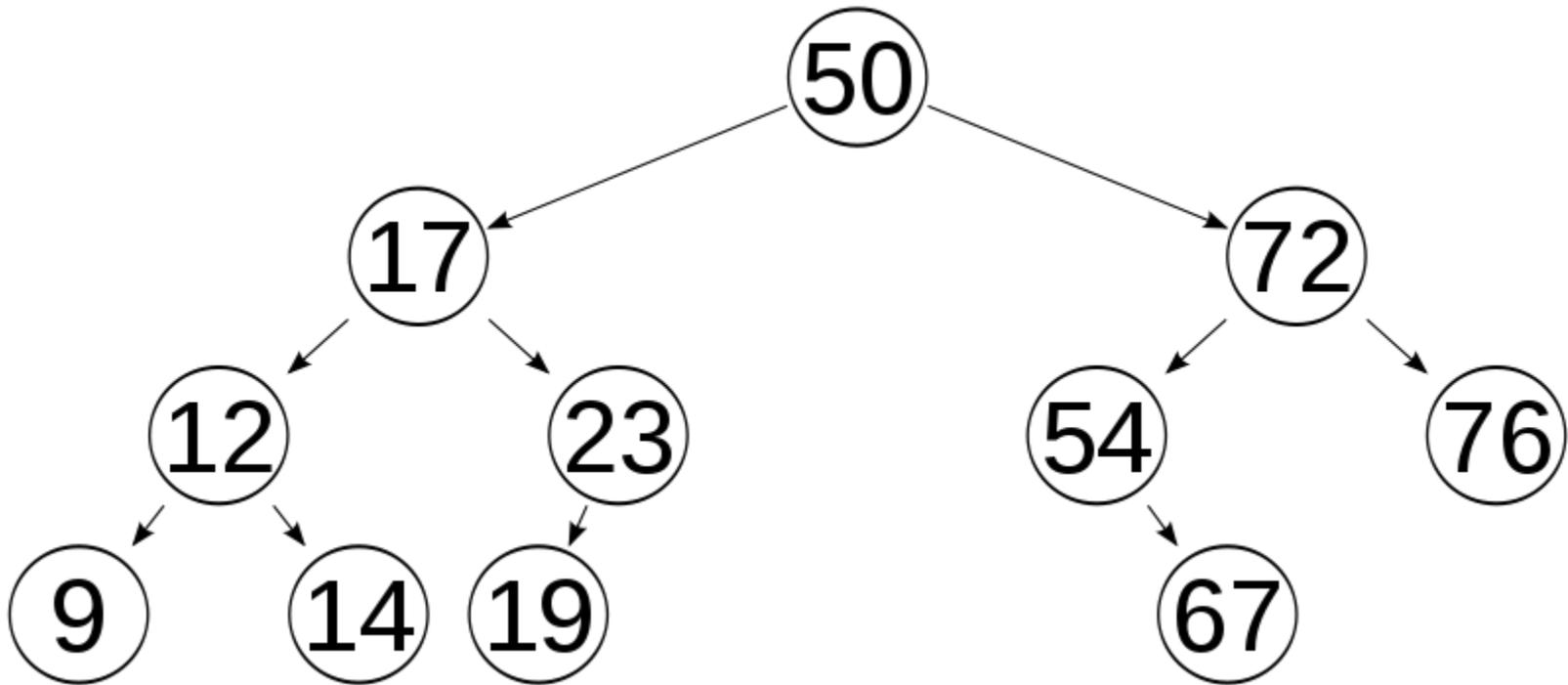
Tema 5 Árboles
5.3. Árboles AVL

Índice

- **Introducción (conceptos básicos)**
- TAD Árbol Binario
 - Recorridos
 - Implementación
- TAD Árbol Binario de Búsqueda
- **Equilibrado de árboles: árboles AVL**

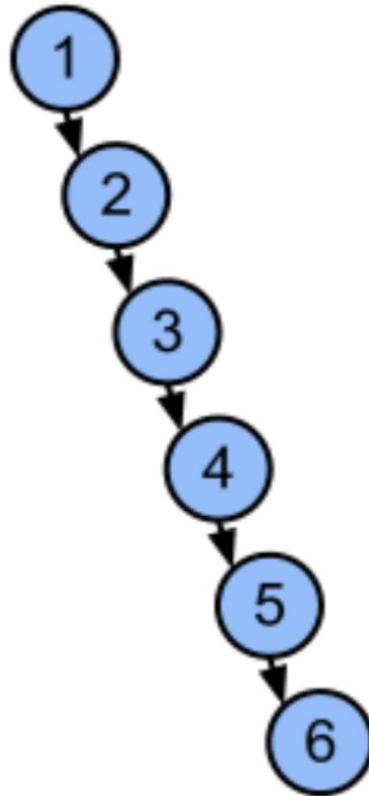
Resumen - ABB

Ventaja: Insert, remove y search $\sim O(h) = O(\log_2 n)$.



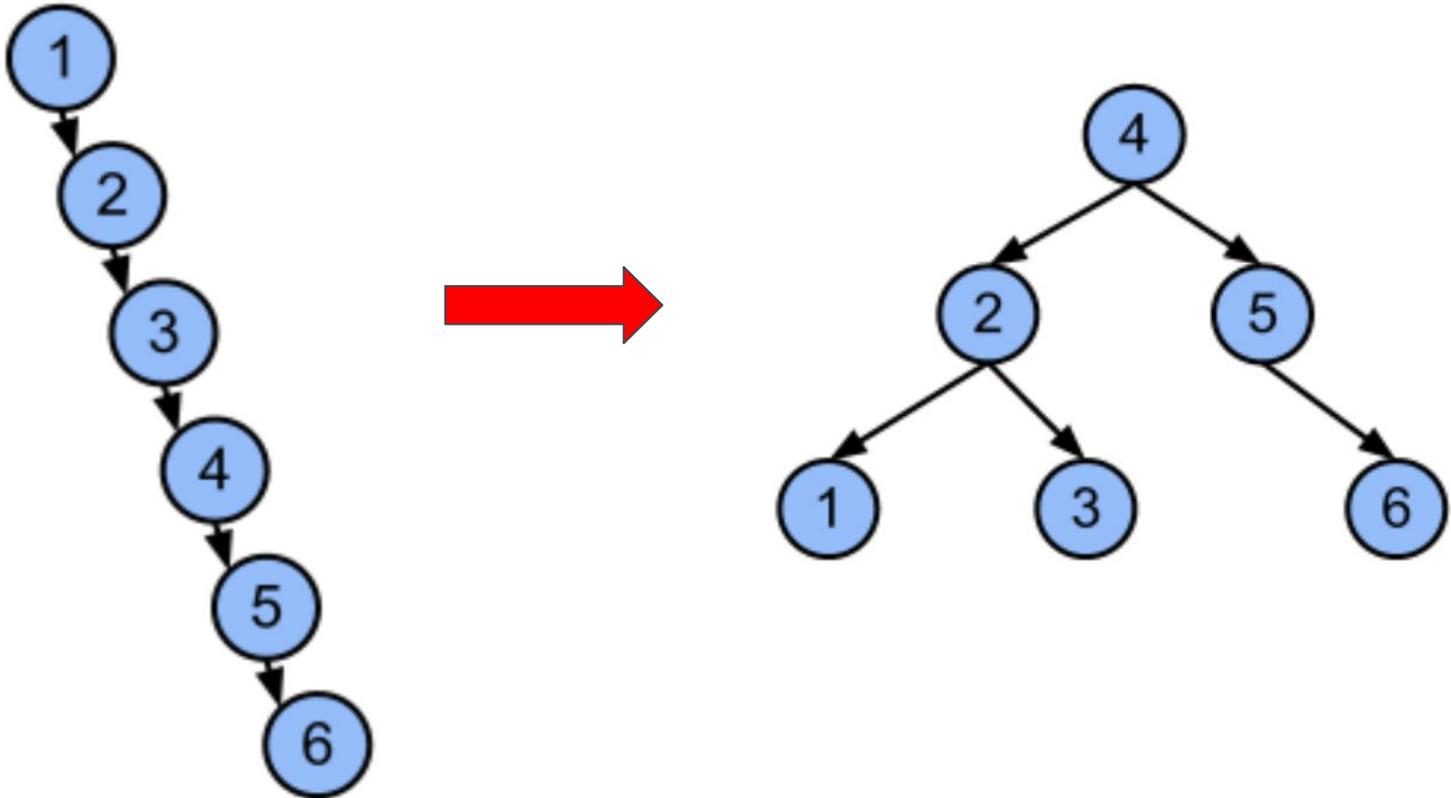
Resumen - ABB

Si h (height) $\approx n \Rightarrow O(h) = O(n)$.



Resumen - ABB

Solución: Equilibrado de un ABB

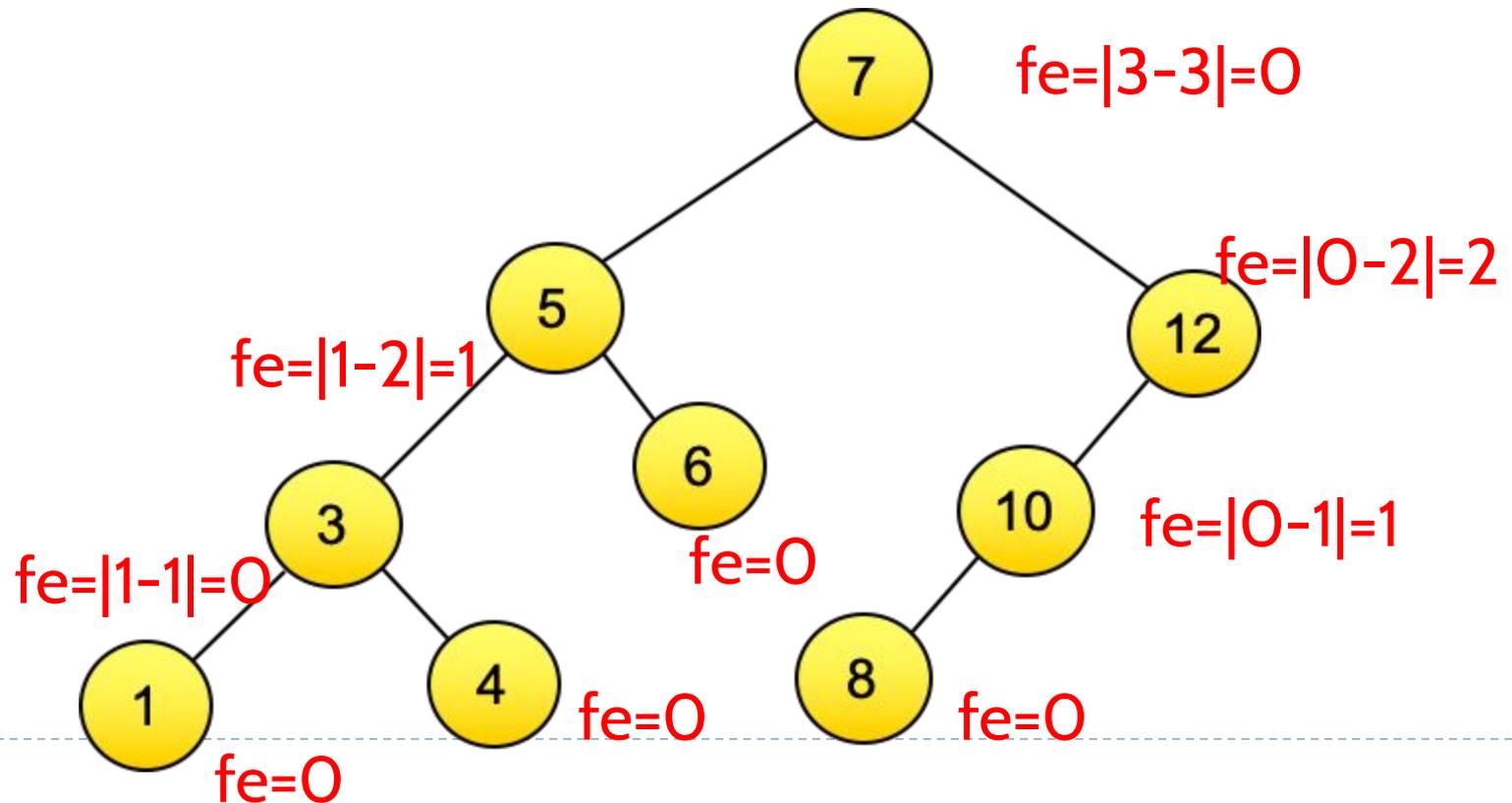


¿Cómo mantener un ABB equilibrado?

- Equilibrado basado en altura (AVL trees, propuestos por los matemáticos Adelson-Velskii y Landis)

Equilibrado basado en altura

Factor de equilibrio (fe) : valor absoluto de la diferencia entre la altura del subárbol derecho y el subárbol izquierdo. $fe = |h_r - h_l|$

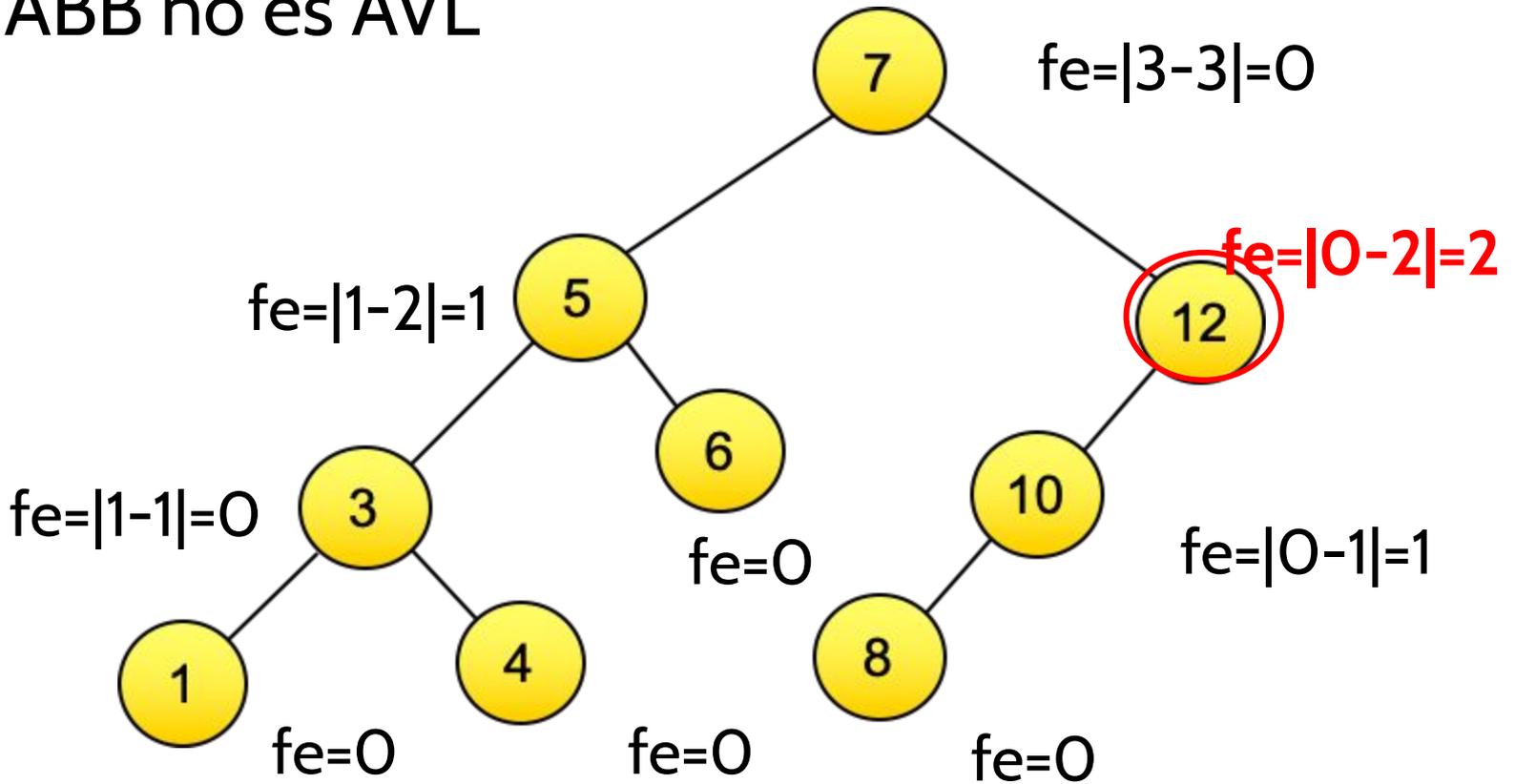


Equilibrado basado en altura (AVL)

- Un árbol es AVL (o está equilibrado en altura), cuando todos sus nodos tienen un factor de equilibrio menor o igual que 1.

Equilibrado basado en altura

Este ABB no es AVL



Equilibrado basado en altura (AVL)

Idea:

Después de cada operación de inserción o borrado, se comprueba si algún nodo ha quedado desbalanceado, es decir, ($fe \geq 2$).

La idea es rotar la rama más larga, para mover nodos de la rama más larga a la rama más corta.

El árbol resultante debe seguir siendo un ABB y estar equilibrado en altura (es decir, ser AVL)

Equilibrado basado en altura

Rotaciones:

Derecha simple

Izquierda simple

Doble rotación (left-right)

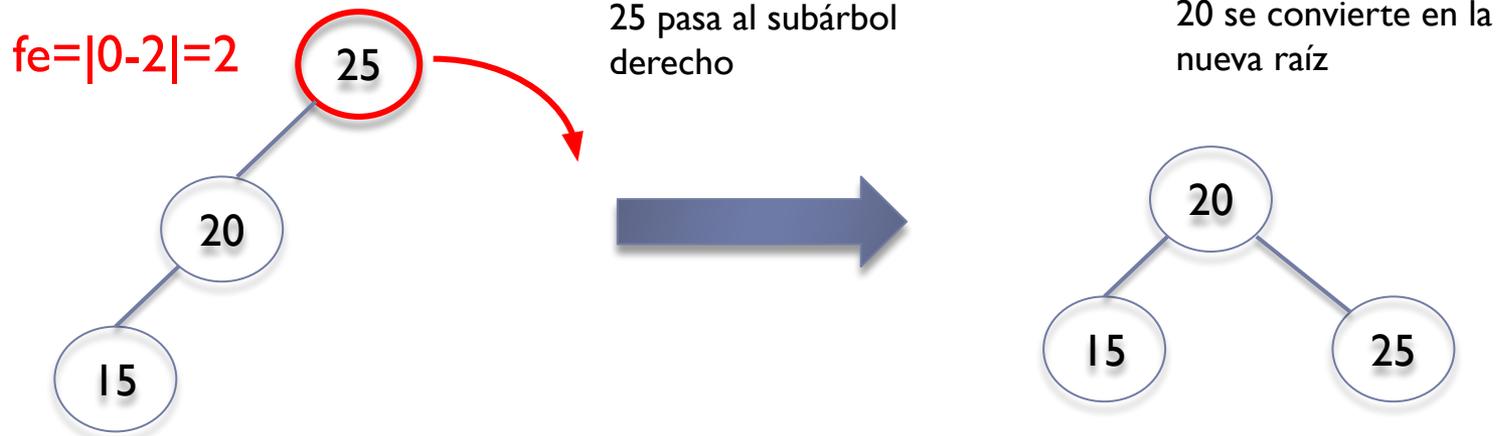
Doble rotación (right-left)

Equilibrado basado en altura

La herramienta *visualgo* te permite crear árboles AVL y ver cómo tras las operaciones de inserción y borrado, qué rotaciones se aplican para que el árbol resultante siga estando equilibrado.

<https://visualgo.net/en/bst>

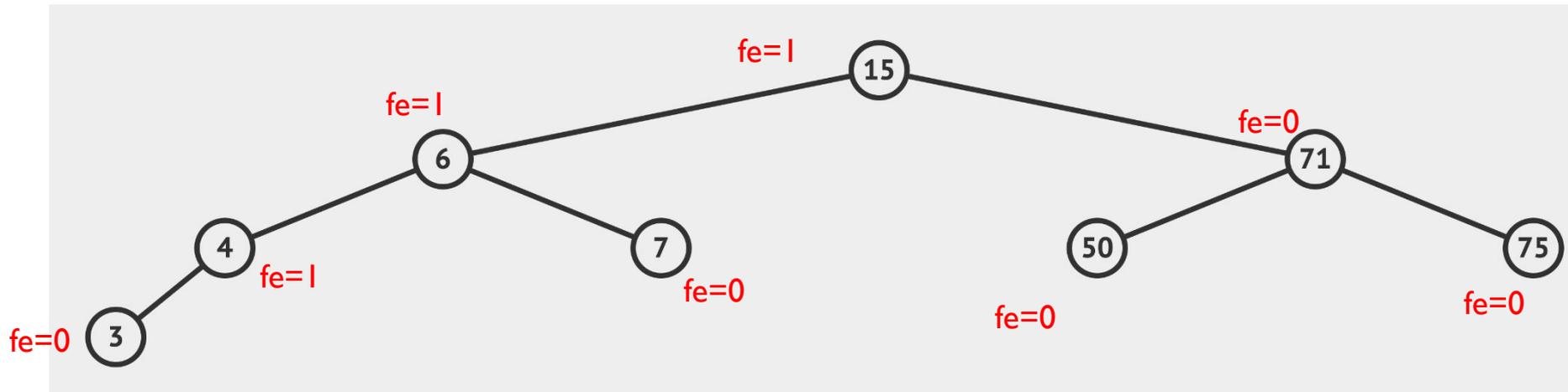
Rotación Simple Derecha



Como la rama más larga es la rama izquierda, rotamos el nodo desbalanceado (25) hacia la derecha. Es decir, el nodo 25 debe pasar al subárbol derecho. La nueva raíz del subárbol será el hijo izquierdo del nodo desbalanceado.

Rotación Simple Derecha

Este árbol está equilibrado.

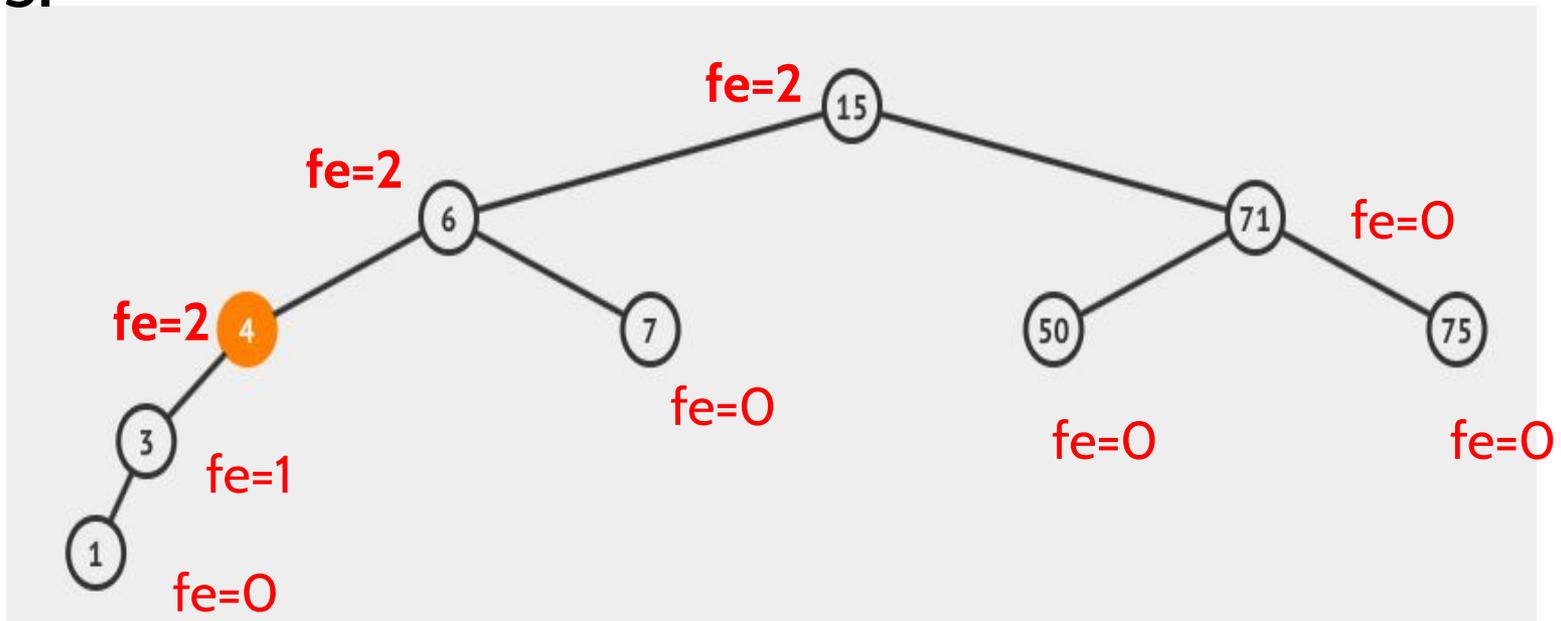


¿Qué ocurre si insertamos 1?



Rotación Simple Derecha

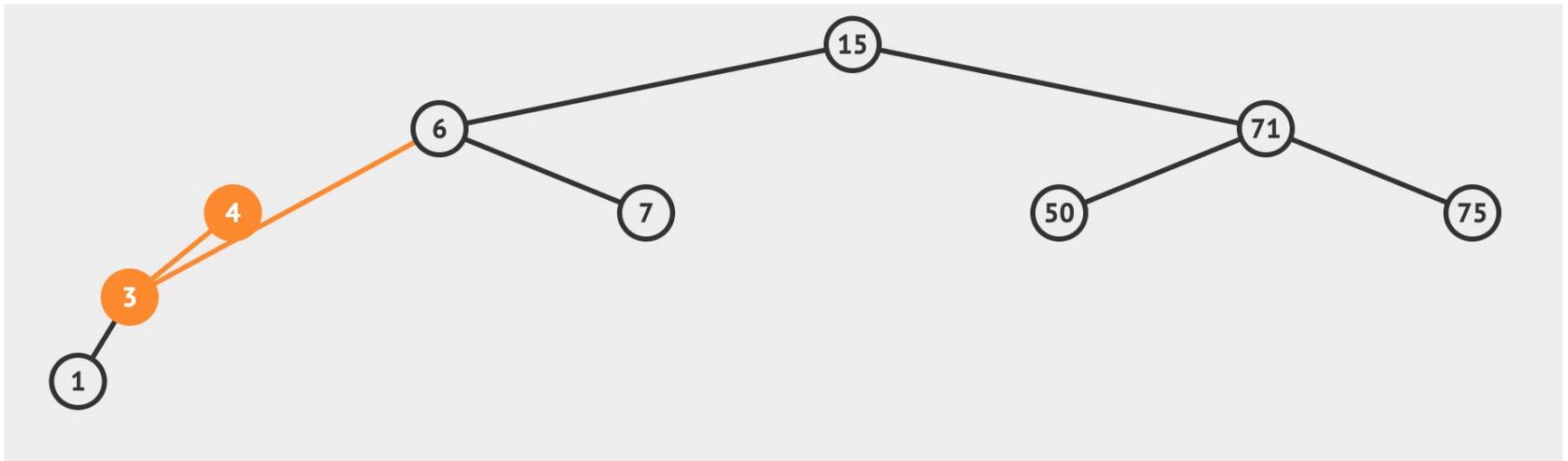
Después de insertar el 1, hay tres nodos no equilibrados: 4, 6 y 15.



El primer nodo que ha quedado desequilibrado es 4. Al equilibrarlo, conseguiremos que queden también equilibrados sus ancestros.

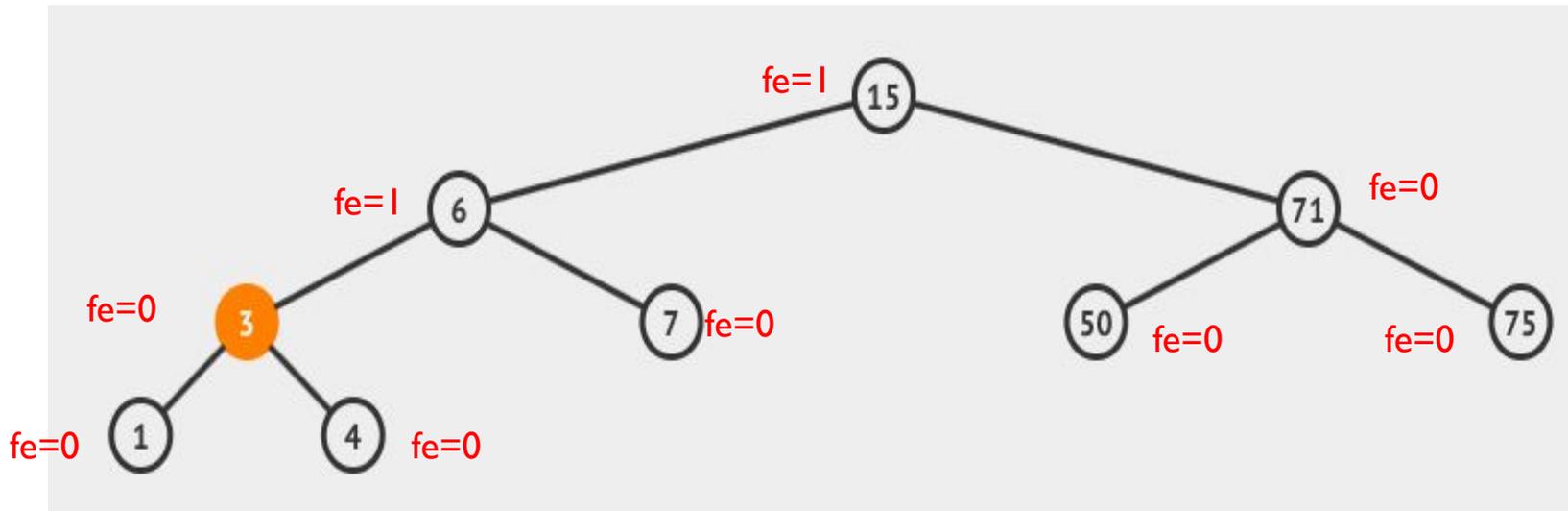
Rotación Simple Derecha

- Considerando el nodo desequilibrado (4), su rama más larga es la izquierda. Para equilibrar demos rotar el nodo (4) a la derecha.
- La nueva raíz será el nodo 3 (que es el hijo izquierdo del nodo 4), y el nodo 4, rotará a la derecha, convirtiéndose en el hijo derecho de la nueva raíz.



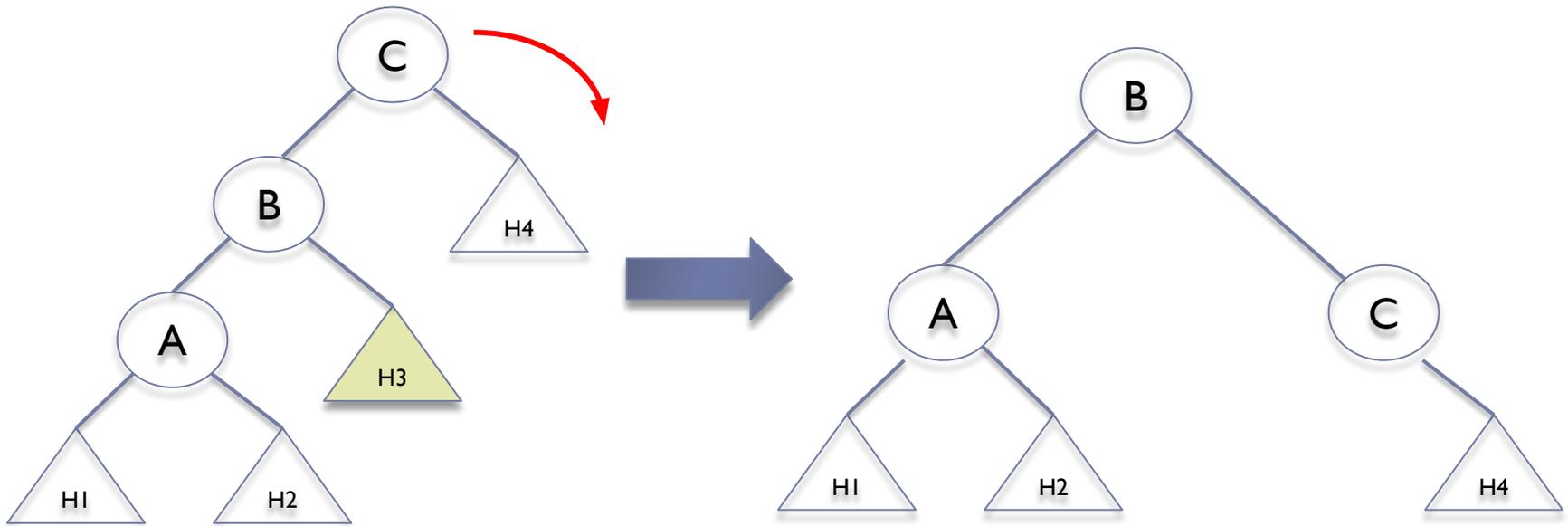
Rotación Simple Derecha

El árbol resultante después de aplicar la rotación derecha es el siguiente (donde todos los nodos están equilibrados):



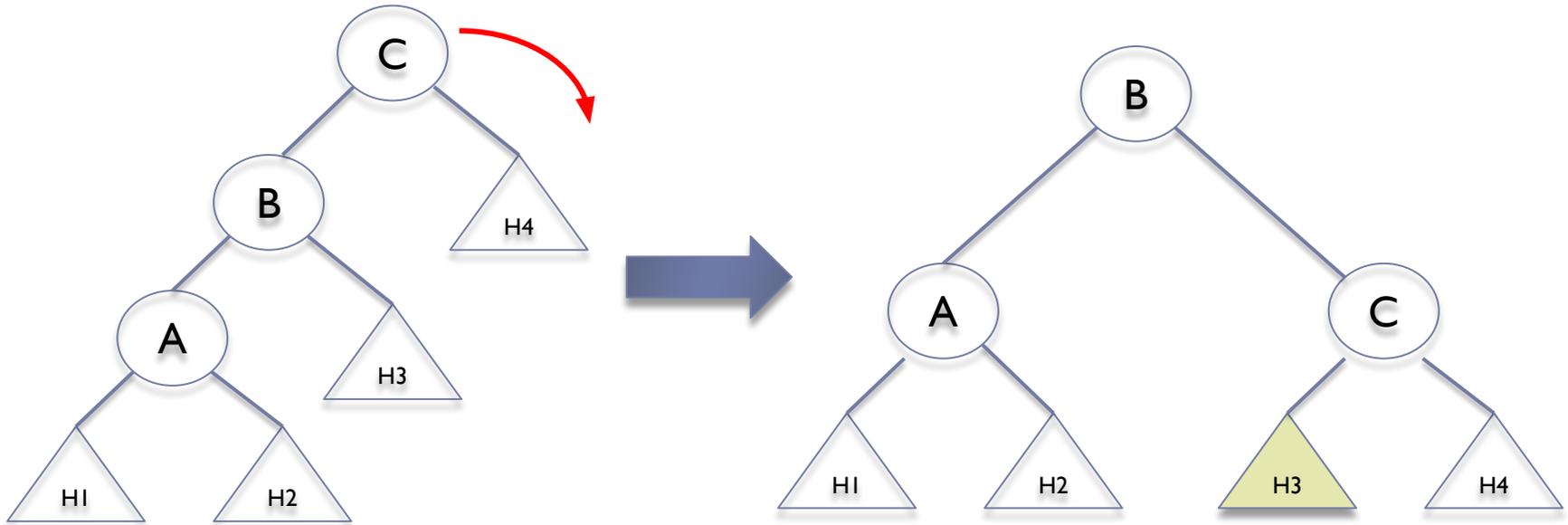
<http://visualgo.net/bst.html>

Rotación Simple Derecha



¿Qué pasa si la nueva raíz (B) tenía subárbol derecho?, El nuevo hijo derecho de B es C, ¿qué hacemos con H3?

Rotación Simple Derecha



InOrder: $H_1 A H_2 B H_3 C H_4$

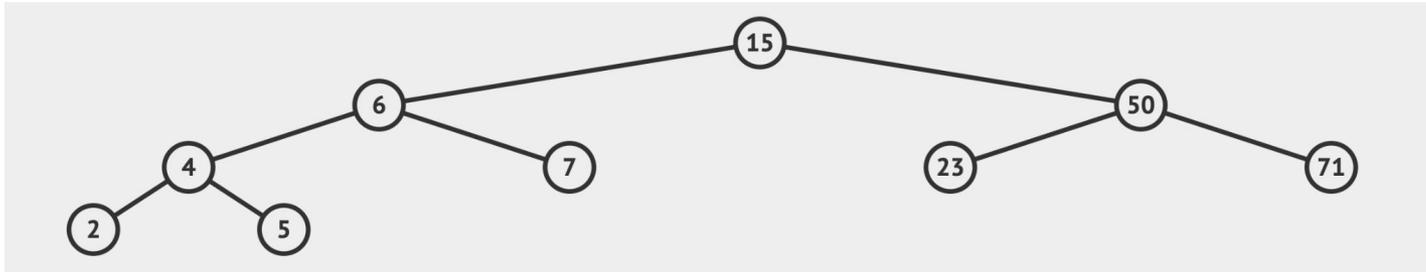
=

InOrder: $H_1 A H_2 B H_3 C H_4$

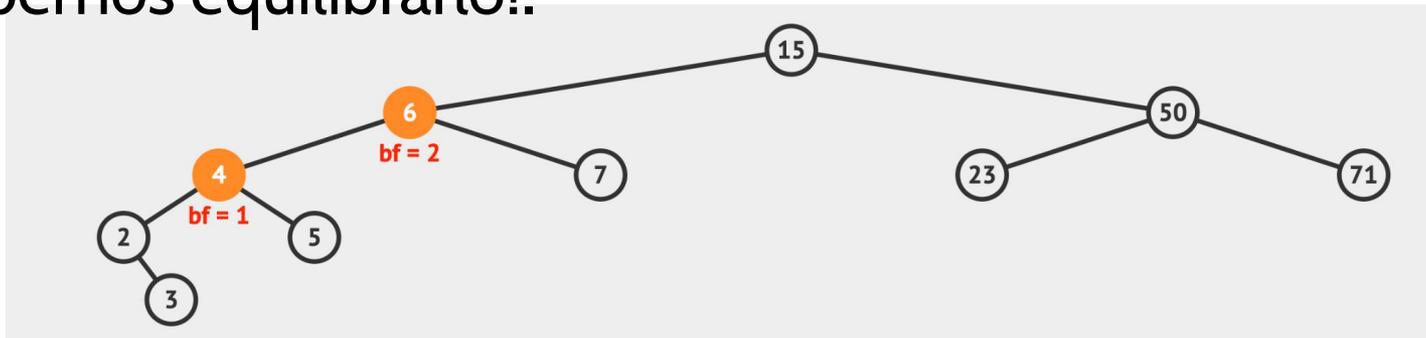
La única opción es que pase a ser hijo izquierdo de C (cualquier otra opción dejaría de ser un ABB)

Rotación Simple Derecha

¿Qué pasa si en el siguiente árbol insertamos el elemento 3?

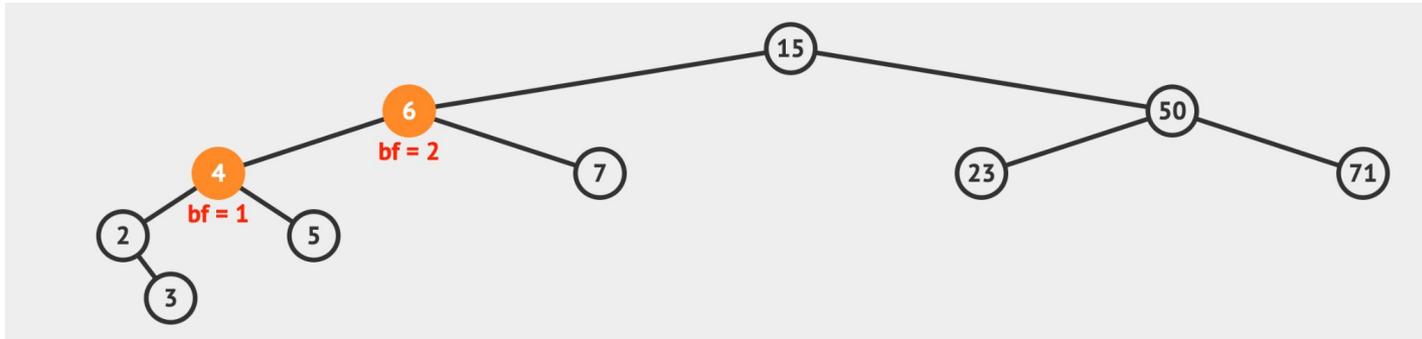


Después de insertar 3, el nodo 6, queda desbalanceado, ¡debemos equilibrarlo!



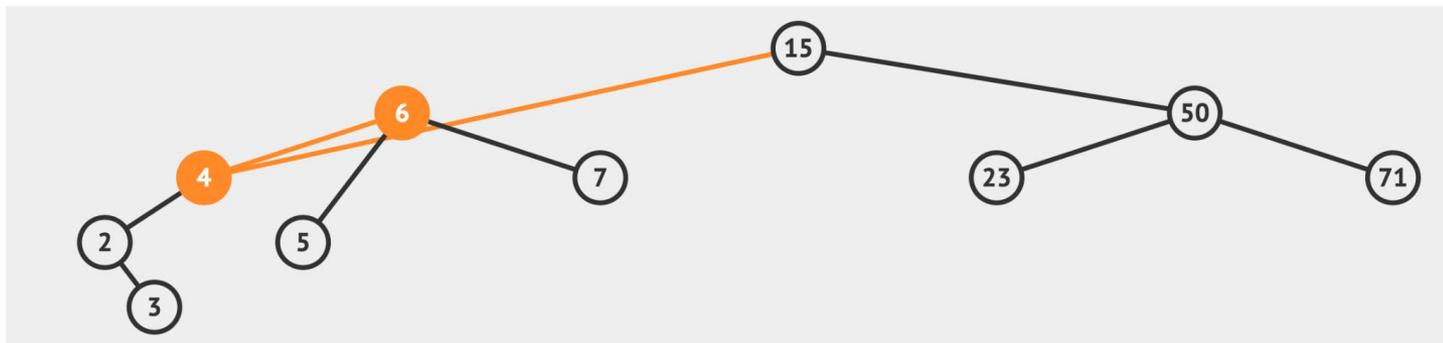
Nota: 15 también queda desequilibrado (bf=2), pero tras equilibrar 6, quedará equilibrado.

Rotación Simple Derecha

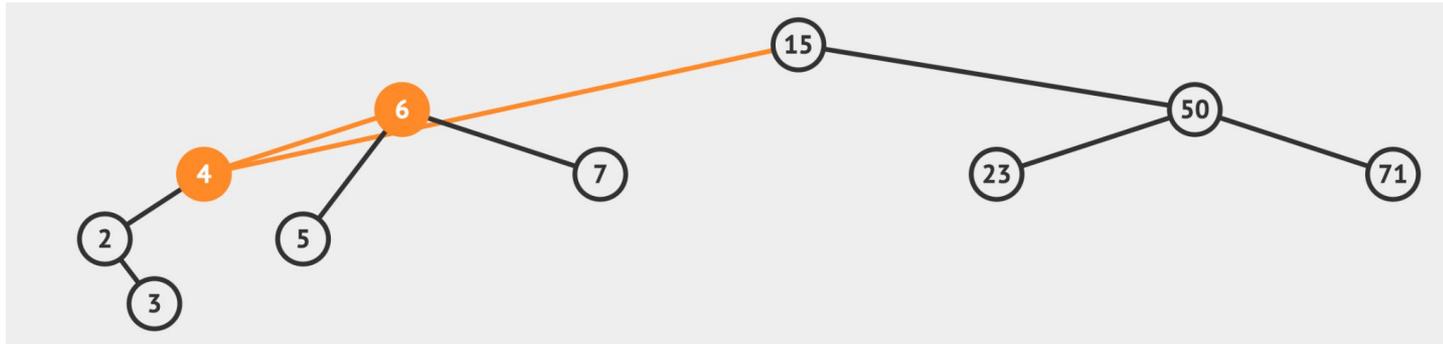


Para equilibrar el nodo 6, como su rama más larga es la izquierda, el nodo 6 deberá rotar a la derecha.

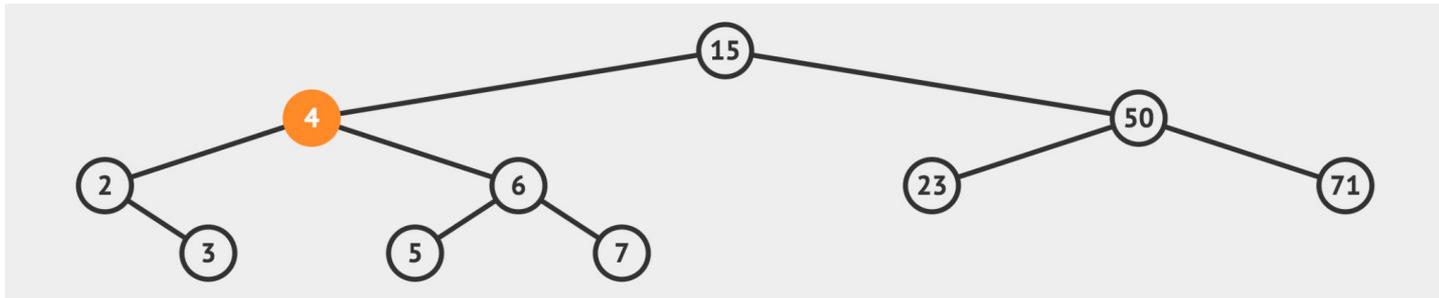
Si aplicamos la rotación simple derecha, el nodo 4, se convertirá en la nueva raíz, y su actual hijo derecho debe



Rotación Simple Derecha

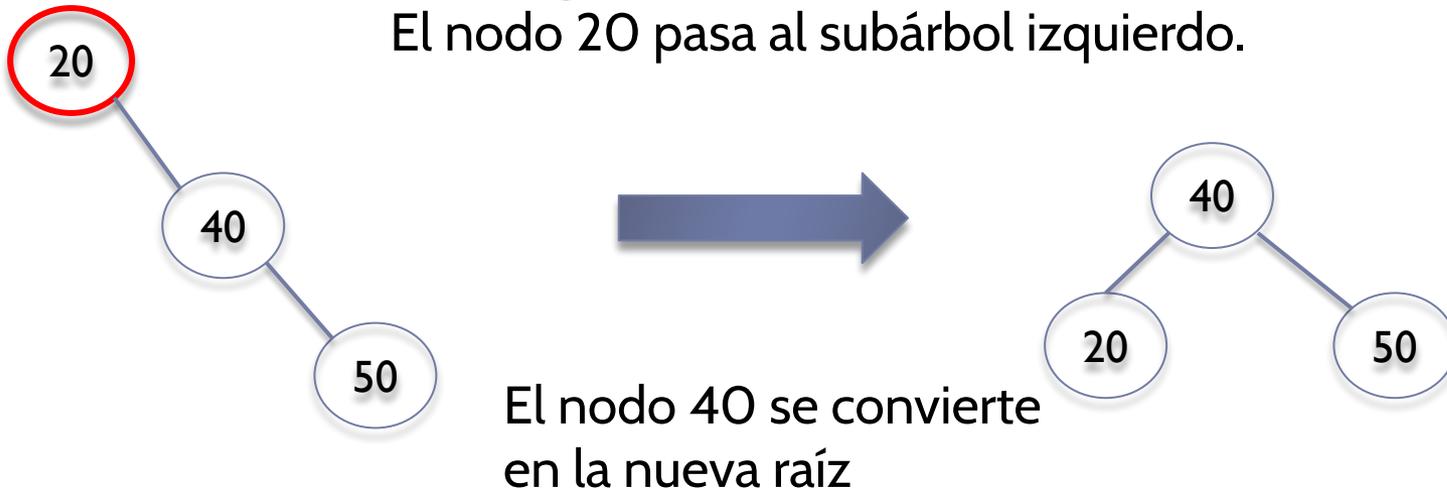


Finalmente, el nodo 6, se convierte en el nuevo hijo derecho, de la nueva raíz del subárbol (nodo 4). En el nuevo árbol resultante, todos los nodos ya están balanceados (también el nodo 15).



Rotación Simple Izquierda

El nodo 20 está desequilibrado. La rama más larga es la derecha.
El nodo 20 pasa al subárbol izquierdo.

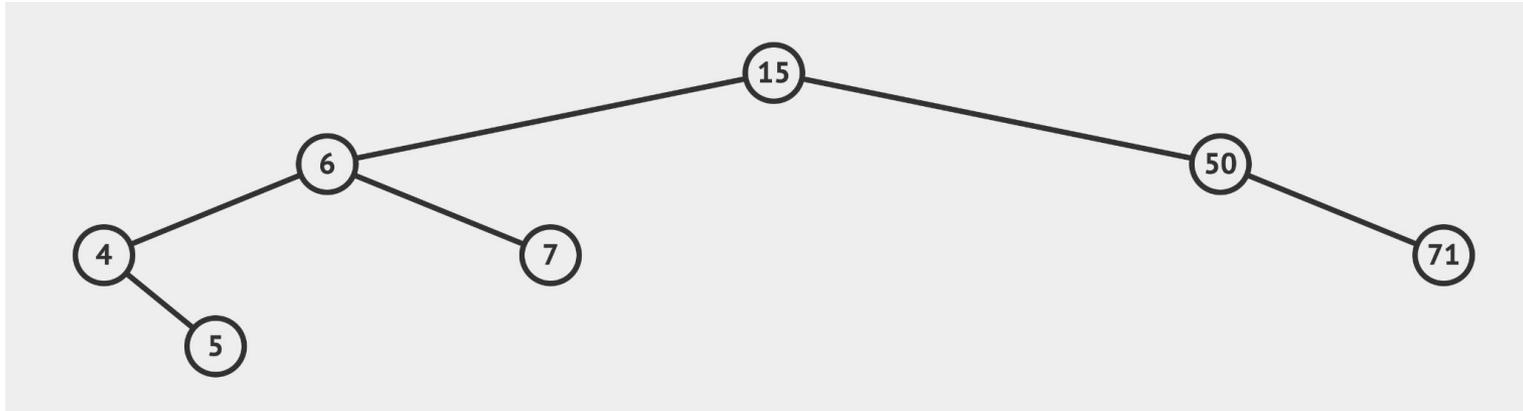


El nodo 40 se convierte en la nueva raíz

Como la rama más larga es la rama derecha, el nodo desbalanceado (20) debe rotar para pasar al subárbol izquierdo. Por eso se denomina rotación izquierda. La nueva raíz del subárbol será el hijo derecho (nodo 40) del nodo desbalanceado.

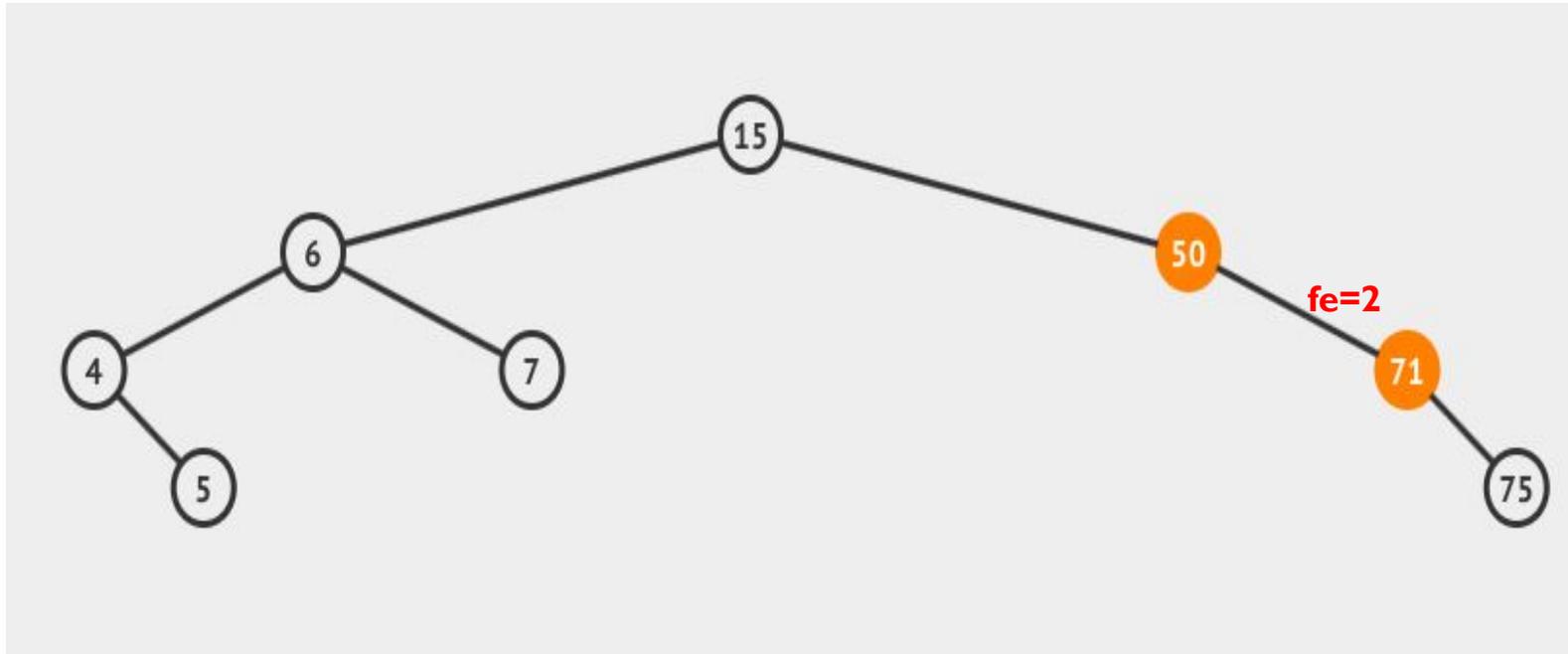
Rotación Simple Izquierda

El árbol está equilibrado. ¿Qué ocurre si insertamos 75?



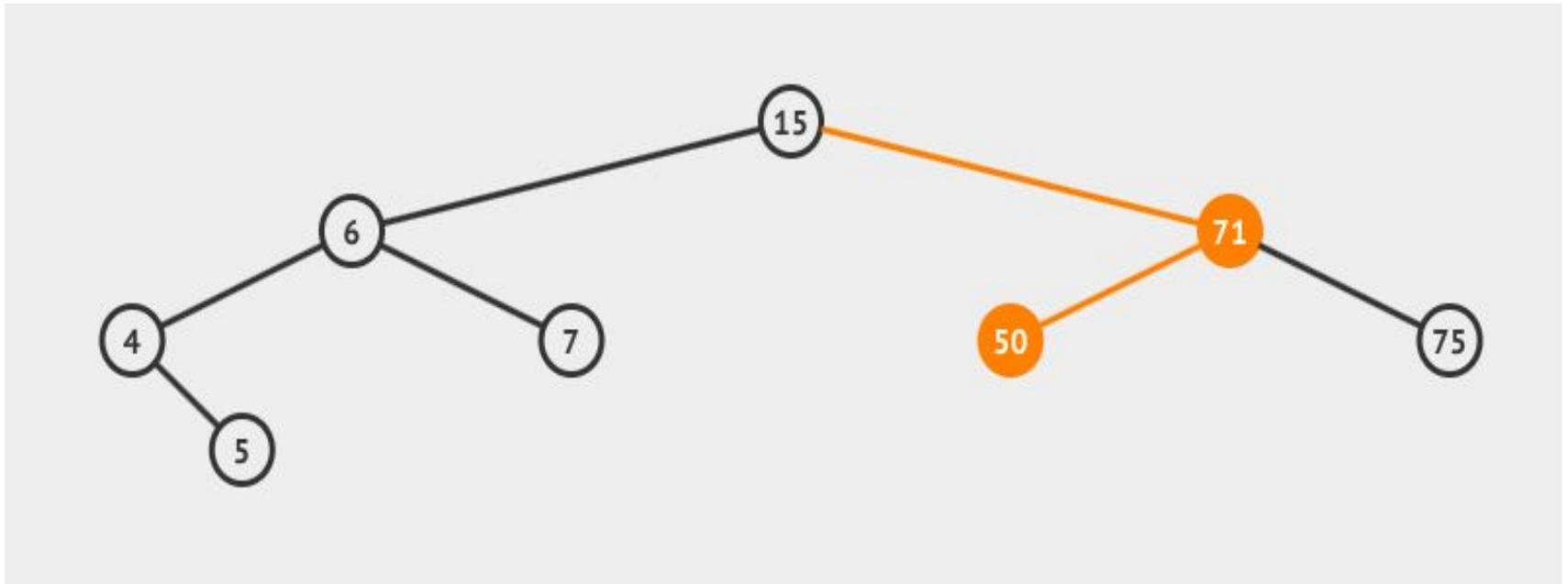
<http://visualgo.net/bst.html>

Rotación Simple Izquierda



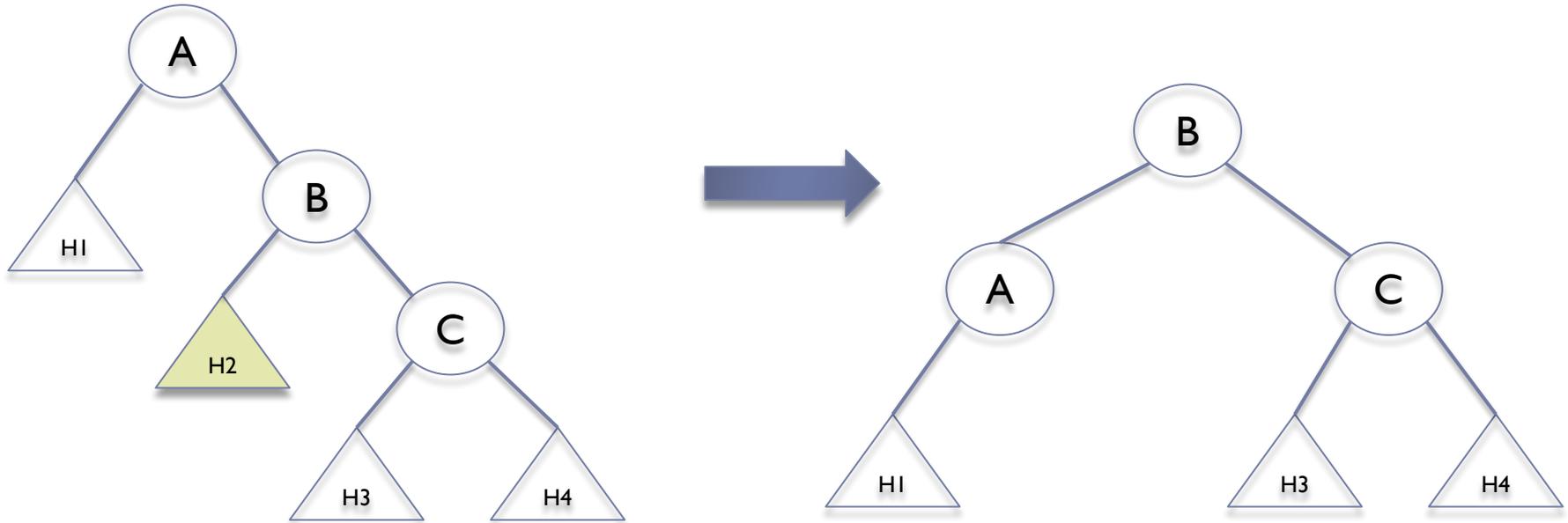
Al insertar 75, 50 queda desequilibrado ($fe=2$). Debemos aplicar una rotación simple izquierda. 71 será la nueva raíz y 50 rotará a la izquierda, convirtiéndose en el nuevo hijo izquierdo de la nueva raíz (50).

Rotación Simple Izquierda



Todos los nodos ya están equilibrados ($fe \leq 1$).

Rotación Simple Izquierda

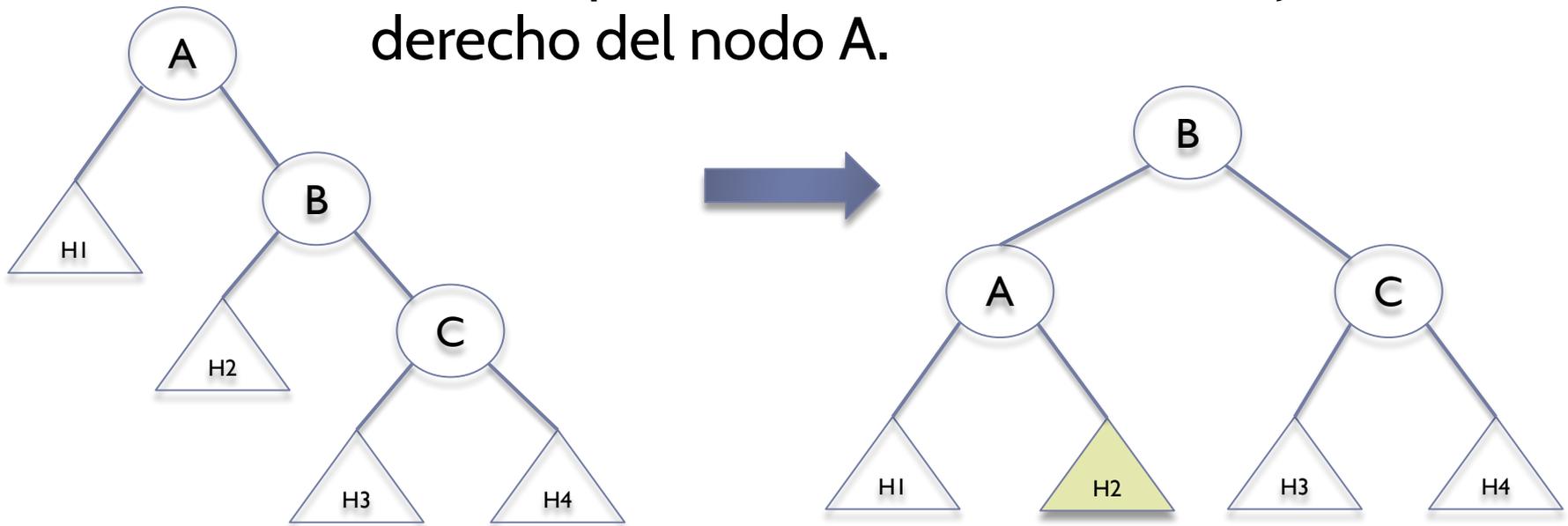


¿Qué si la nueva raíz (B) ya tenía un hijo izquierdo? El nodo desbalanceado (A) al rotar a la izquierda se convierte en el nuevo hijo de la nueva raíz (B).

¿Qué podemos hacer con el subárbol H2, hijo izquierdo de B?

Rotación Simple Izquierda

La única opción para que siga siendo un ABB es que H2 se convierta en el hijo derecho del nodo A.



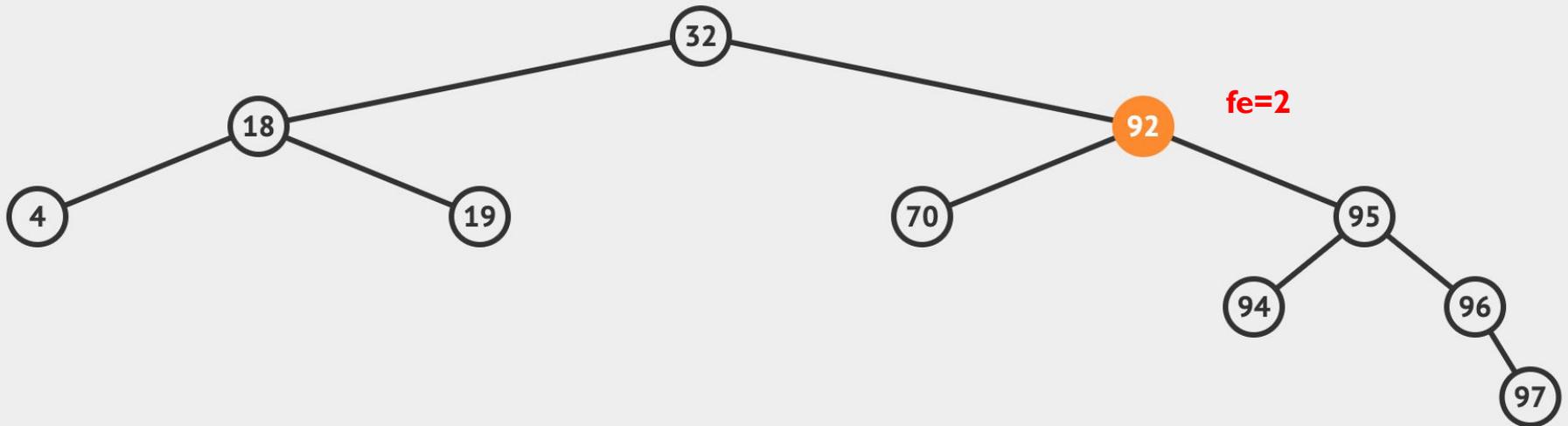
InOrder: $H_1 A H_2 B H_3 C H_4$

=

InOrder: $H_1 A H_2 B H_3 C H_4$

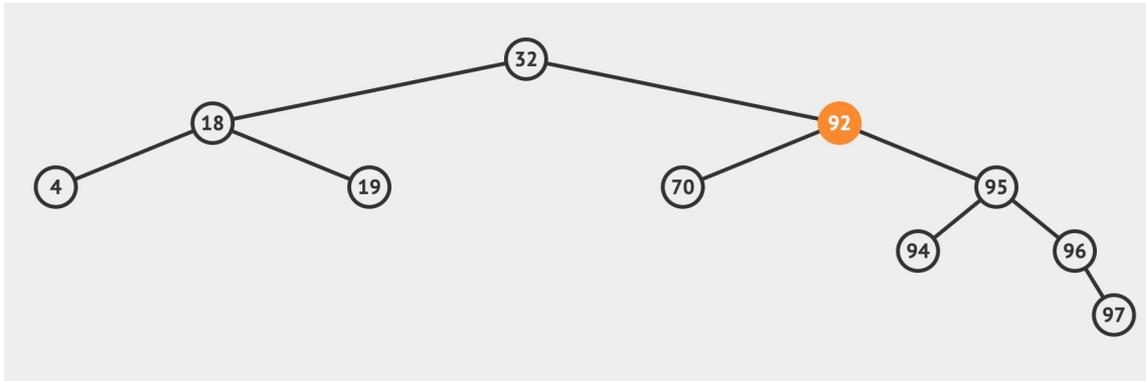
Rotación Simple Izquierda

Supongamos que acabamos de insertar 97. El árbol ha quedado desequilibrado (nodo 92)

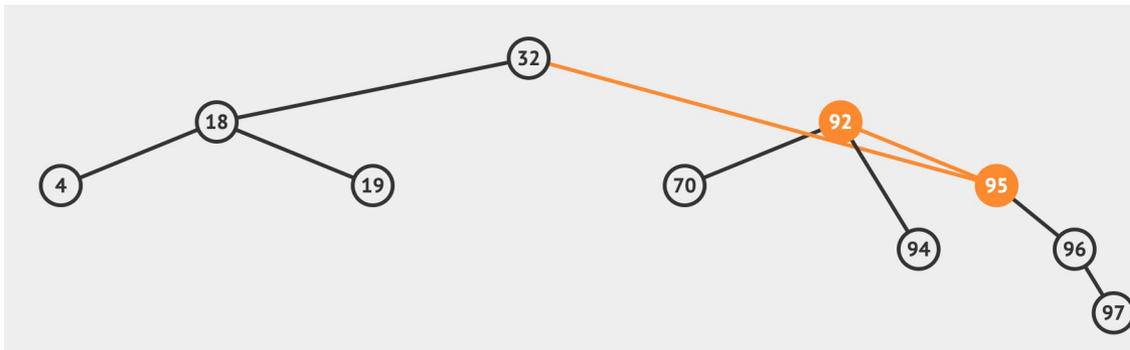


Nota: el nodo raíz (32) también ha quedado desequilibrado ($fe=2$). Después de equilibrar 92, el nodo raíz también quedará equilibrado.

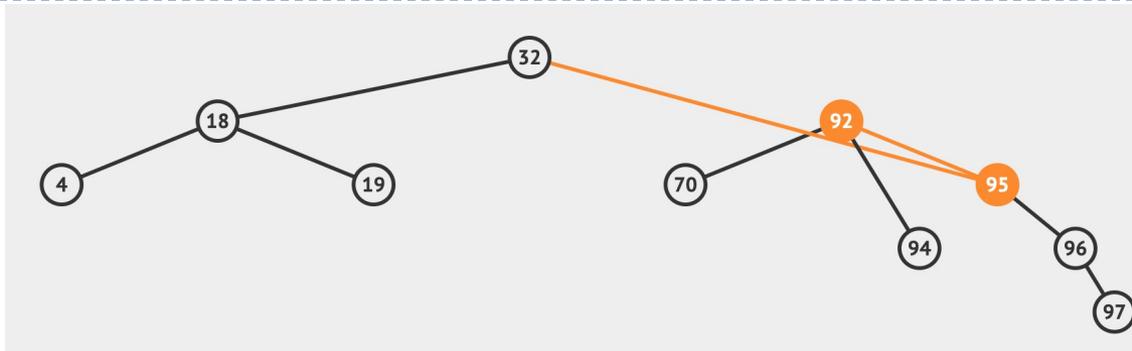
Rotación Simple Izquierda



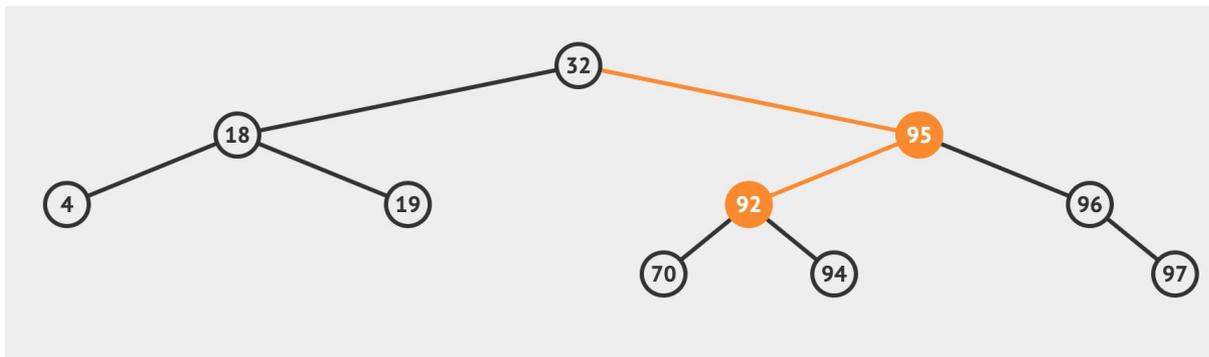
El nodo 92 está desequilibrado. Su rama más larga es la derecha, por tanto, el nodo 92 tiene que rotar a la izquierda. El nuevo nodo raíz del subárbol será su hijo izquierdo: nodo 95. Este nodo, antes de convertirse en la nueva raíz del subárbol, deberá pasar su hijo derecho, 94, para que se convierta en el hijo izquierdo de 92.



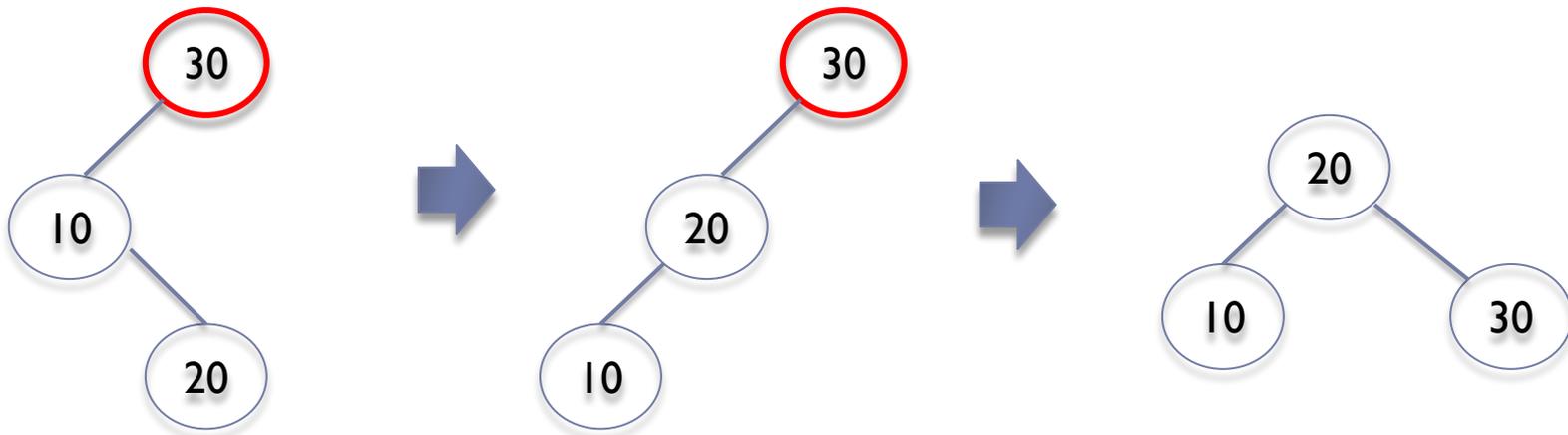
Rotación Simple Izquierda



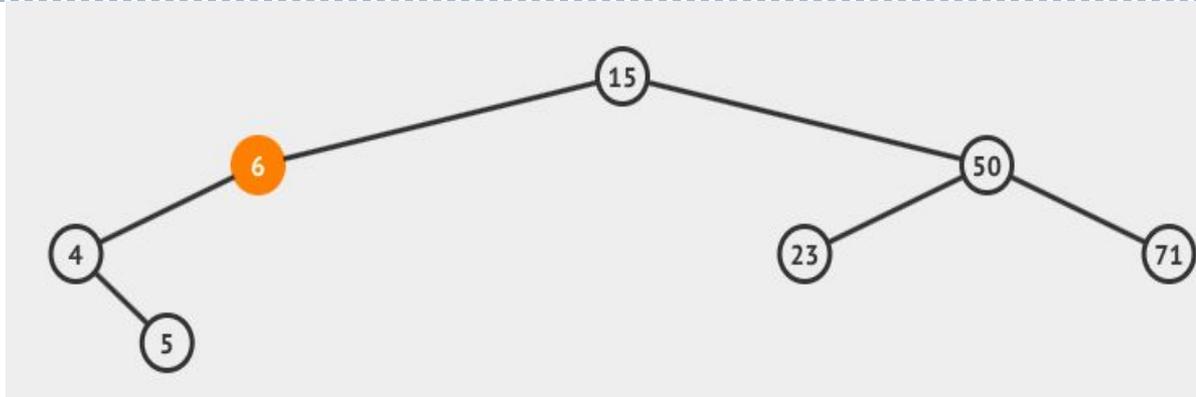
Finalmente, el antiguo nodo desbalanceado 92, se convierte en el nodo izquierdo de la nueva raíz del subárbol, 95. Todos los nodos del árbol ya están balanceados ($fe \leq 1$), incluido la raíz del árbol.



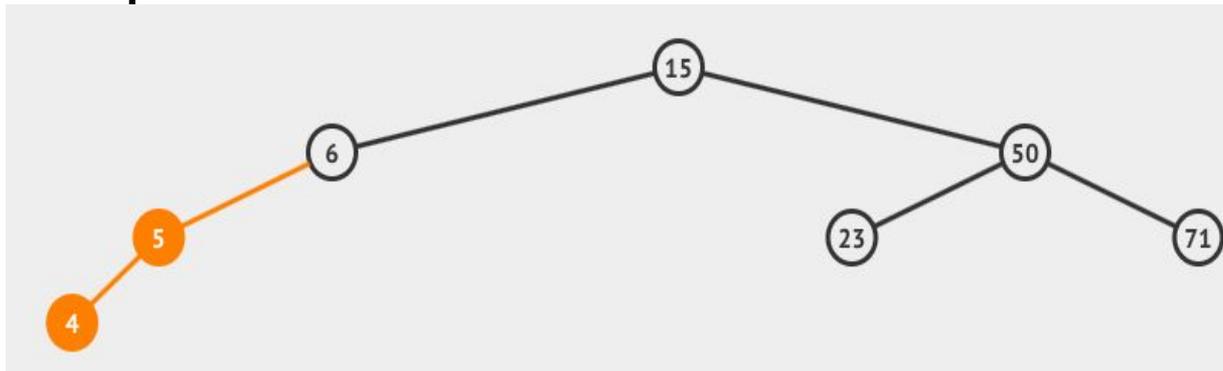
Doble Rotación Izquierda Derecha



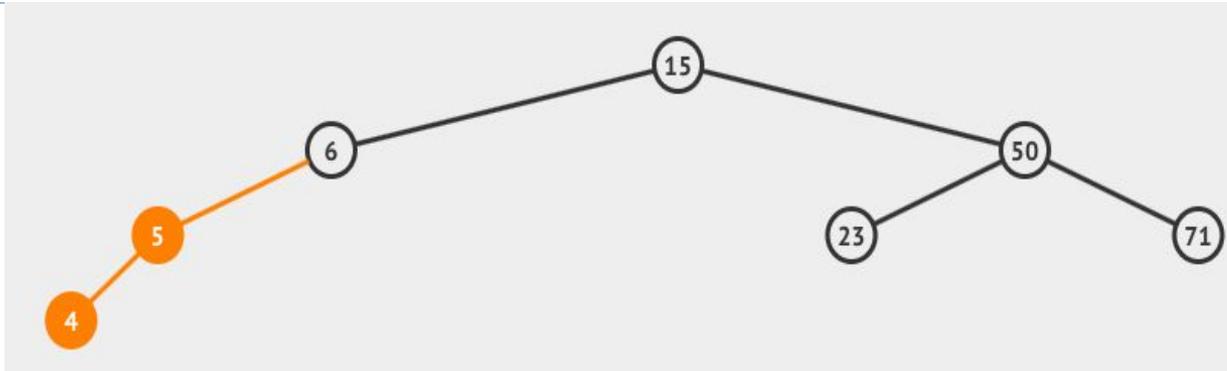
Doble Rotación Izquierda Derecha



El nodo 6 no está equilibrado. Primero debemos rotar el 5 a la izquierda, para que se convierta, en el hijo izquierdo de 6.



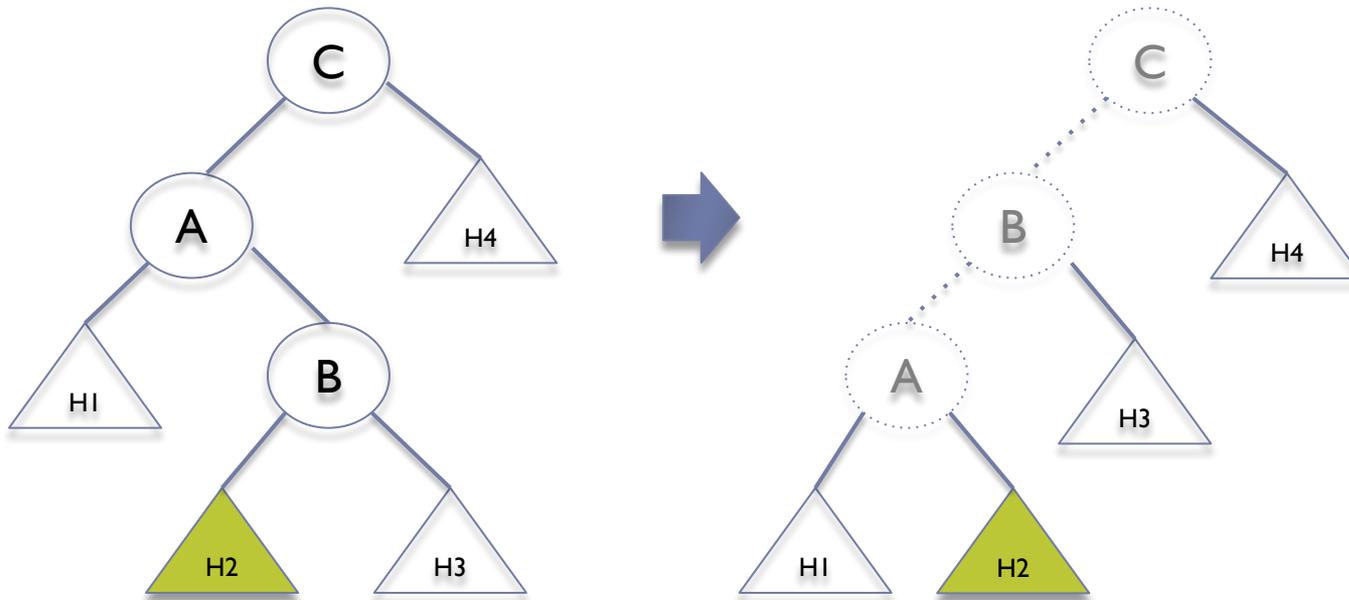
Doble Rotación Izquierda Derecha



Después de haber realizado es rotación a la izquierda, el nodo 6, sigue desequilibrado, pero ya simplemente tenemos que aplicar una rotación a la derecha. El árbol resultante habrá quedado equilibrado.

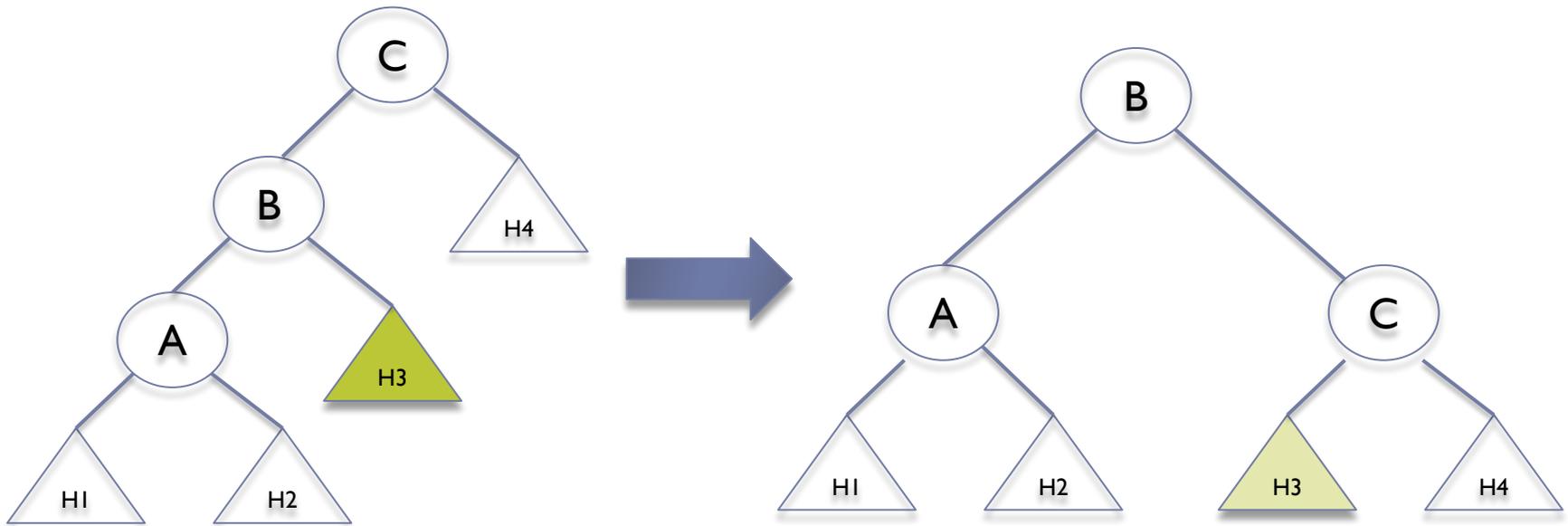


Doble Rotación Izquierda Derecha



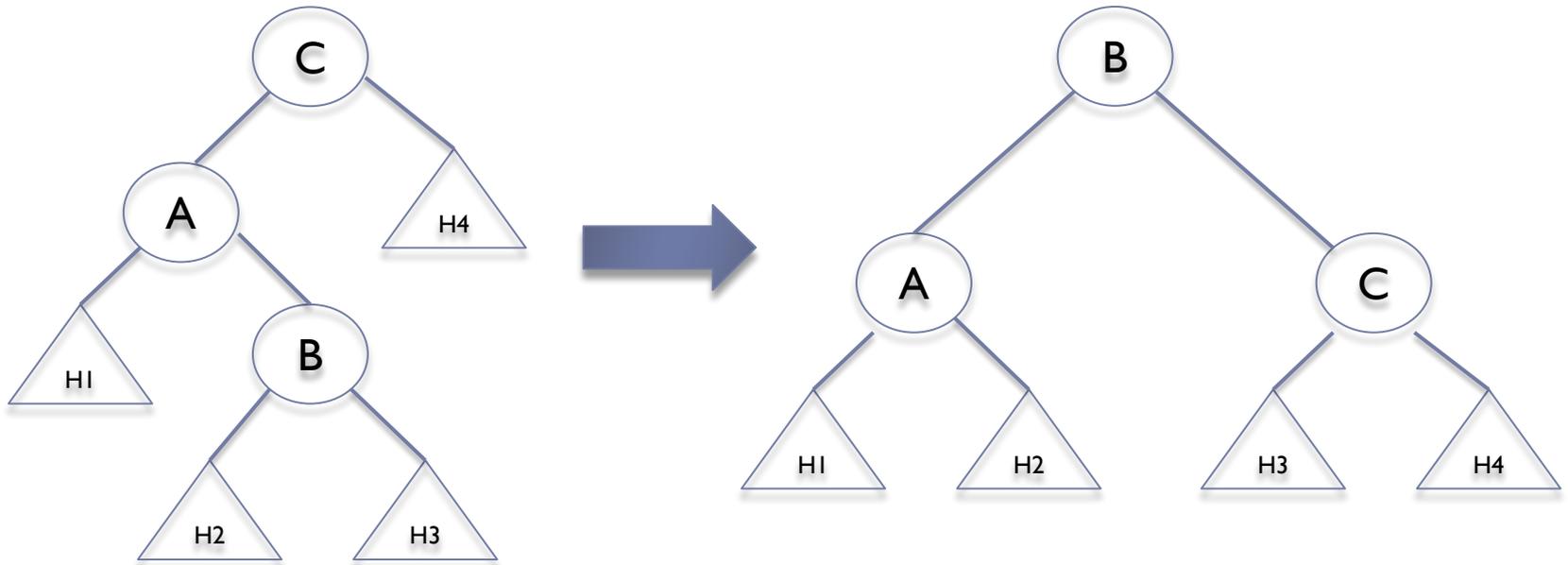
Primer paso, rotación izquierda: rotación de B como hijo izquierdo de C y padre de A. ¿Qué hacemos con sus subárboles? Su subárbol derecho no hay problema, pero su subárbol izquierdo tiene que pasar a ser hijo derecho de A

Doble Rotación Izquierda Derecha



Segunda rotación, rotación derecha: B se convierte a ser la nueva raíz, tomando como nuevo hijo derecho a C. El antiguo hijo derecho de B tiene que pasar a ser hijo izquierdo de C.

Doble Rotación Izquierda Derecha

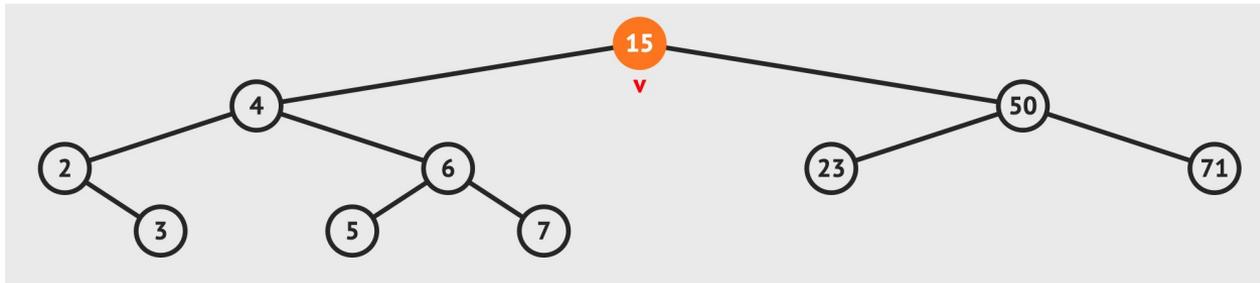


InOrder: $H_1 A H_2 B H_3 C H_4$

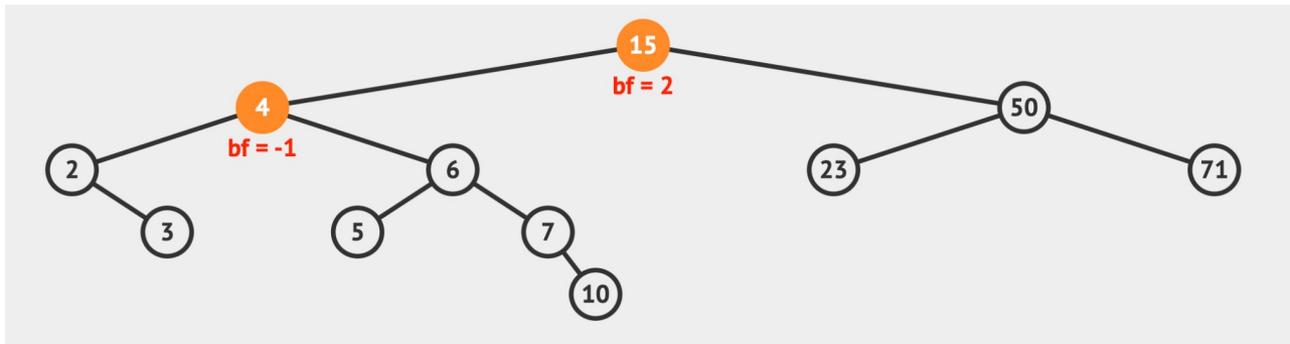
=

InOrder: $H_1 A H_2 B H_3 C H_4$

Doble Rotación Izquierda Derecha

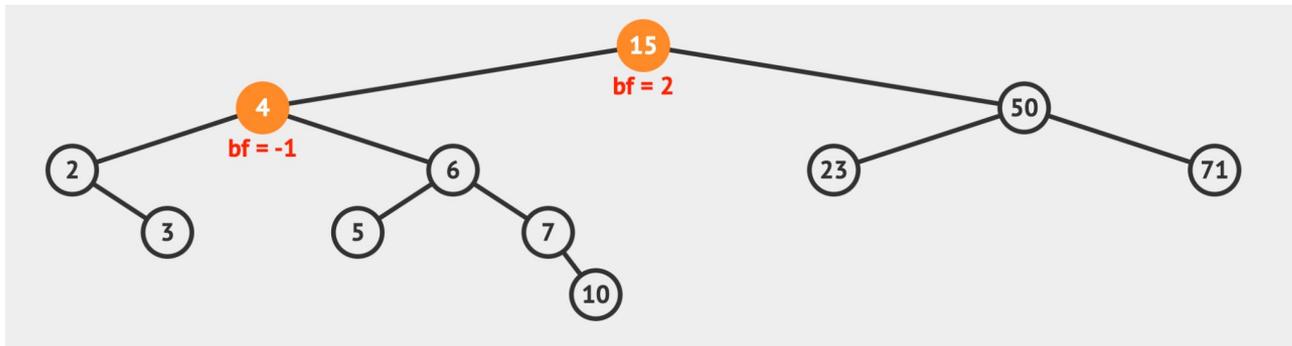


Si insertamos el 10, la raíz no está balanceada:

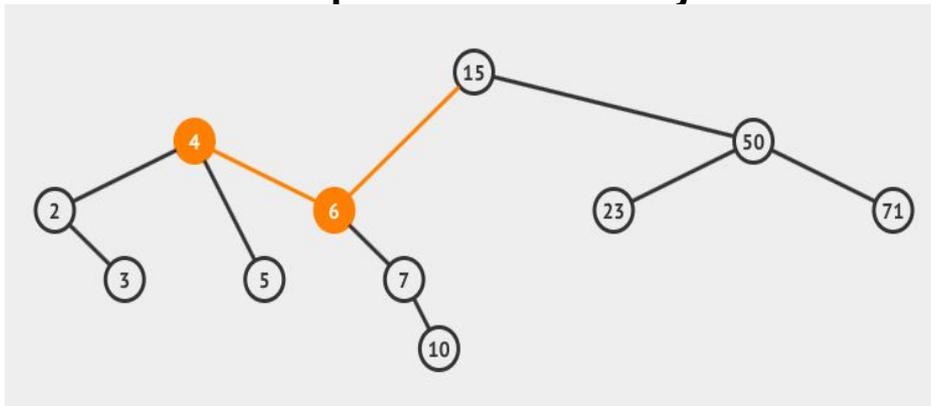


Su rama más larga viene por la izquierda (hasta el 4), y luego por la derecha (nodo 6). Por tanto, deberemos aplicar una rotación izquierda-derecha.

Doble Rotación Izquierda Derecha



Primera rotación: rotación izquierda. El nodo 6 pasa a ser temporalmente la nueva raíz. Su nuevo hijo izquierdo será el nodo 4. Como el nodo 6 ya tiene hijo izquierdo, este debe pasar a ser hijo derecho del nodo 4.

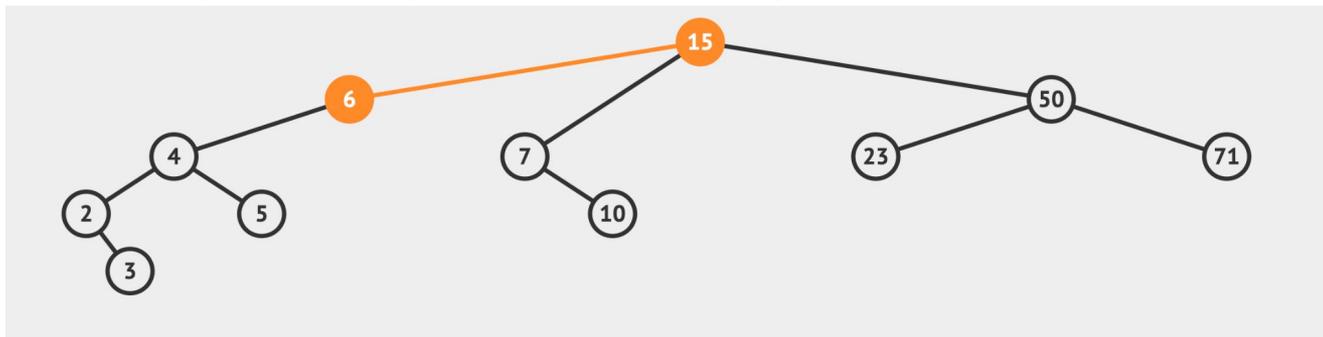


Doble Rotación Izquierda Derecha



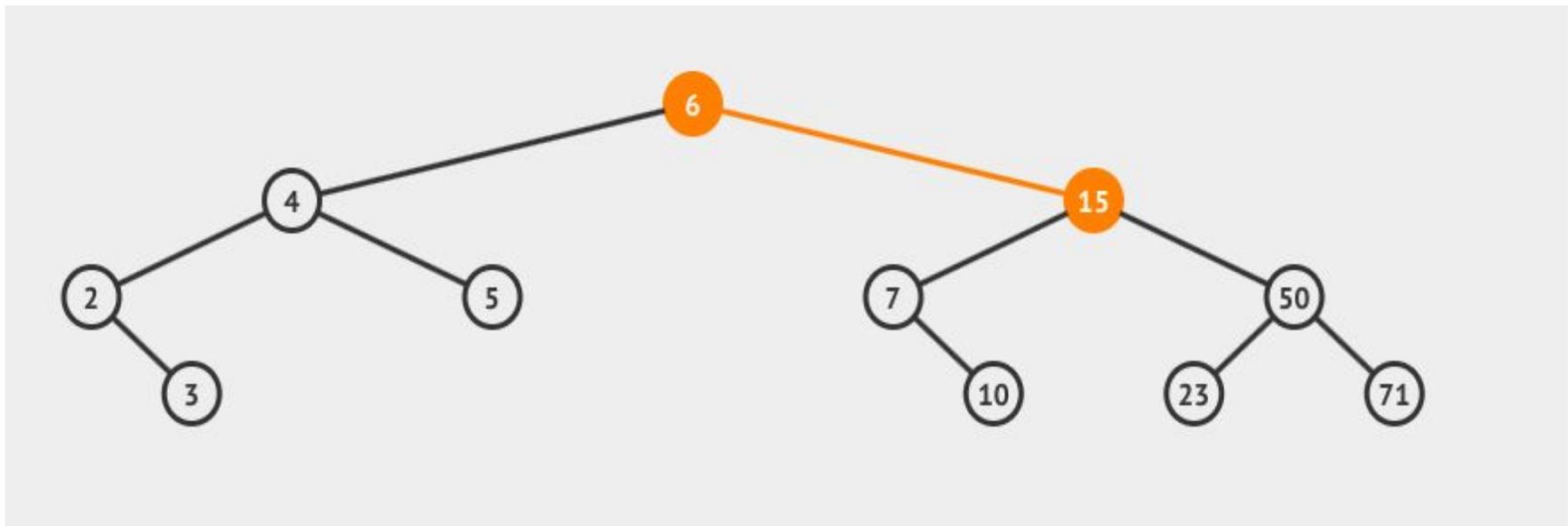
Segunda rotación: rotación derecha. El nodo desequilibrado (sigue siendo el nodo 15), deberá rotar a su subárbol derecho, y su hijo izquierdo se convertirá en el nueva raíz.

Pero antes de hacer eso, el nodo que será la nueva raíz, 6, tiene hijo derecho, el nodo 7. Ese hijo derecho deberá pasar a ser hijo izquierdo de 15. Date cuenta que 15 no puede tener 3 hijos!!!

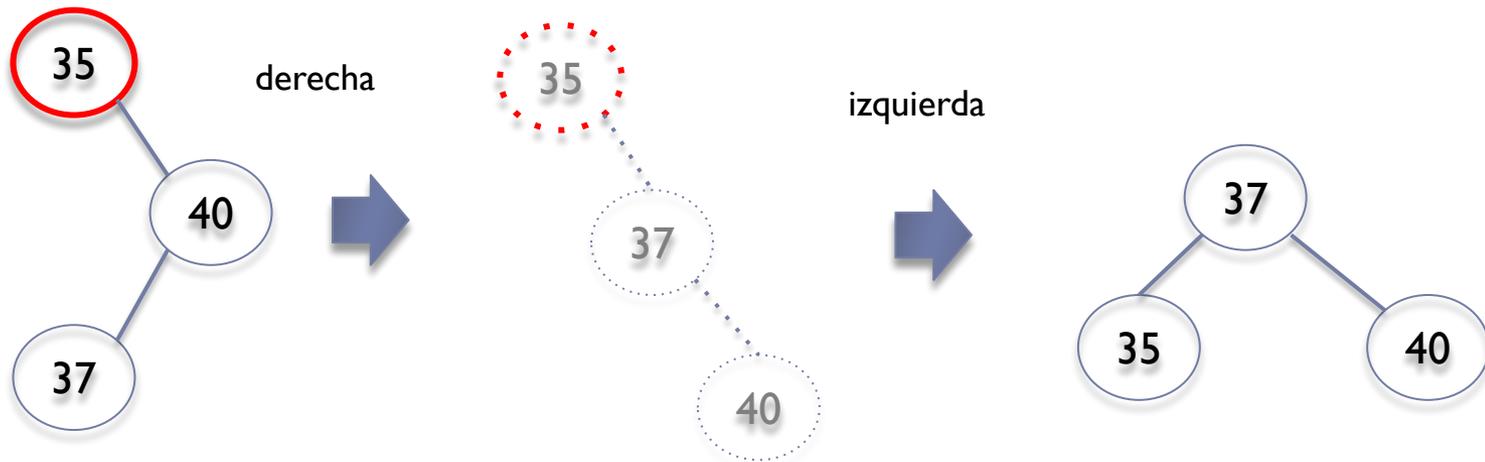


Doble Rotación Izquierda Derecha

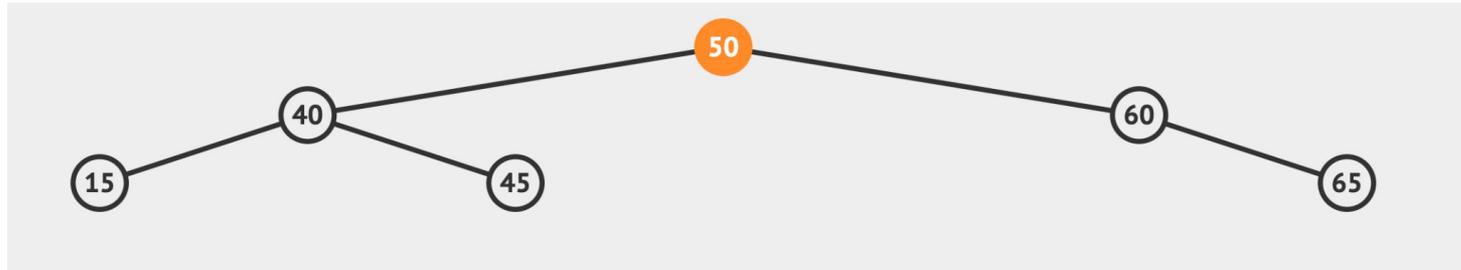
Finalmente, el nodo 6, ya se convierte en la nueva raíz, y el nodo 15, pasa a ser su hijo derecho. Este sería el árbol resultante donde todos los nodos están equilibrados ($fe \leq 1$), después de haber aplicado un doble rotación izquierda-derecha.



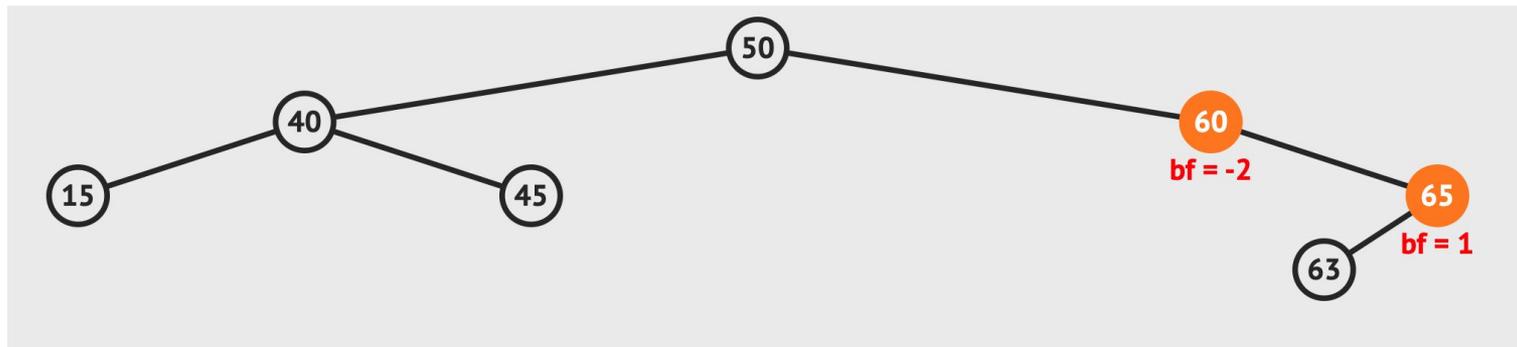
Doble Rotación Derecha Izquierda



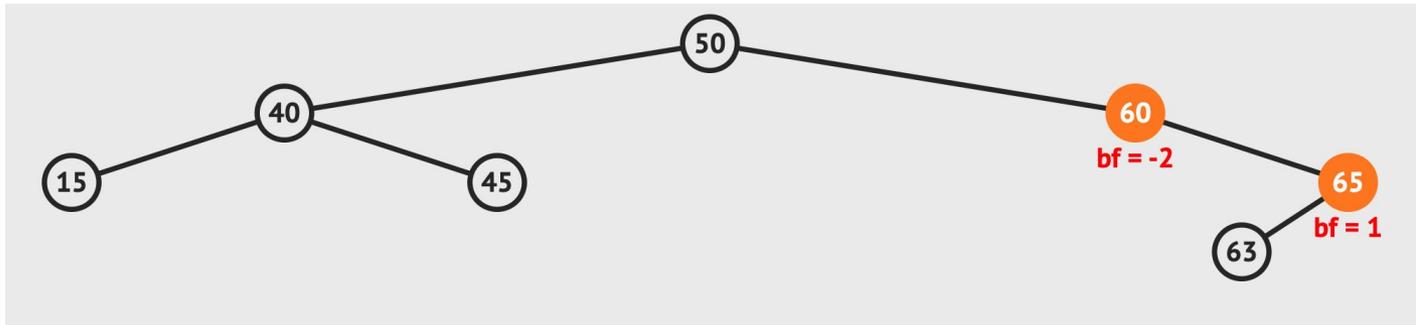
Doble Rotación Derecha Izquierda



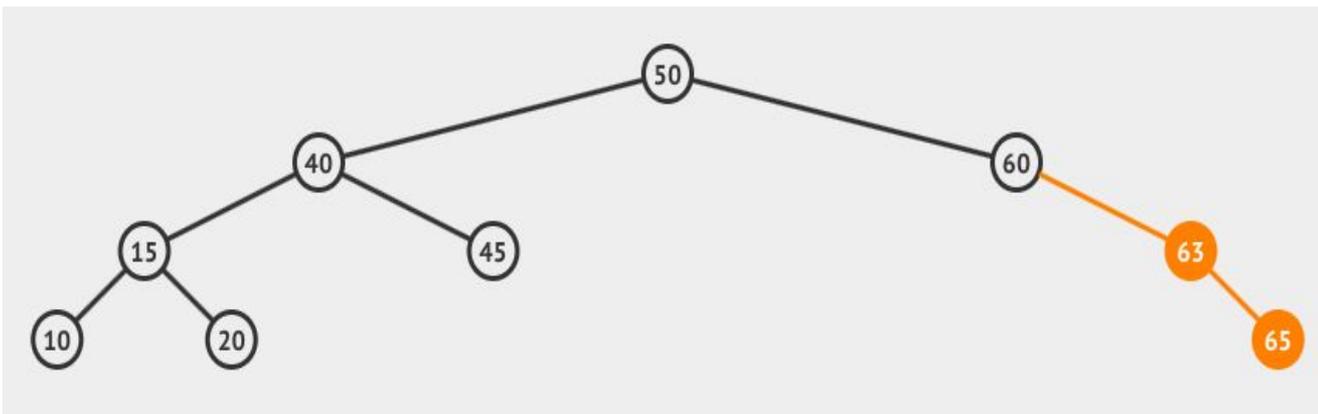
Si insertamos 63, el nodo 60 queda desequilibrado. Debemos aplicar una doble rotación derecha-izquierda



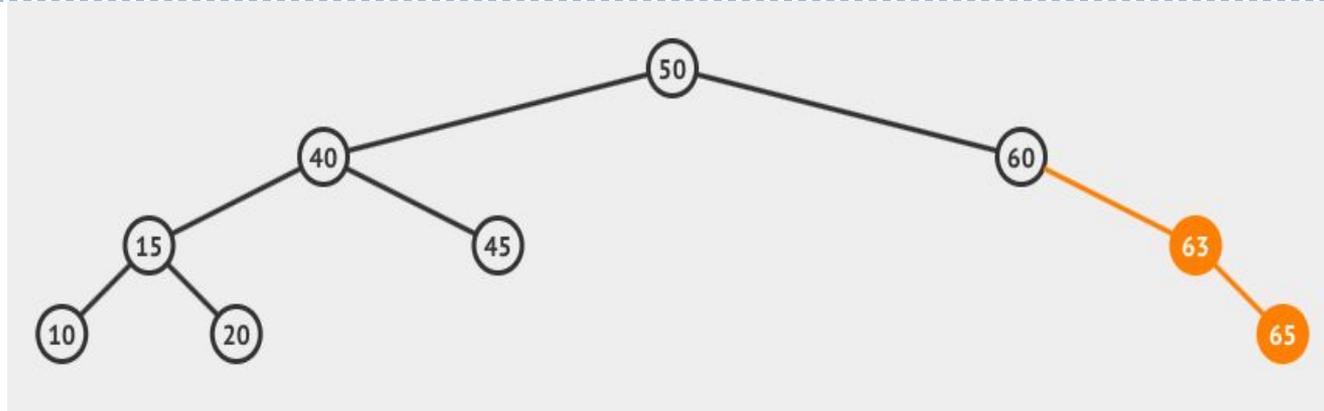
Doble Rotación Derecha Izquierda



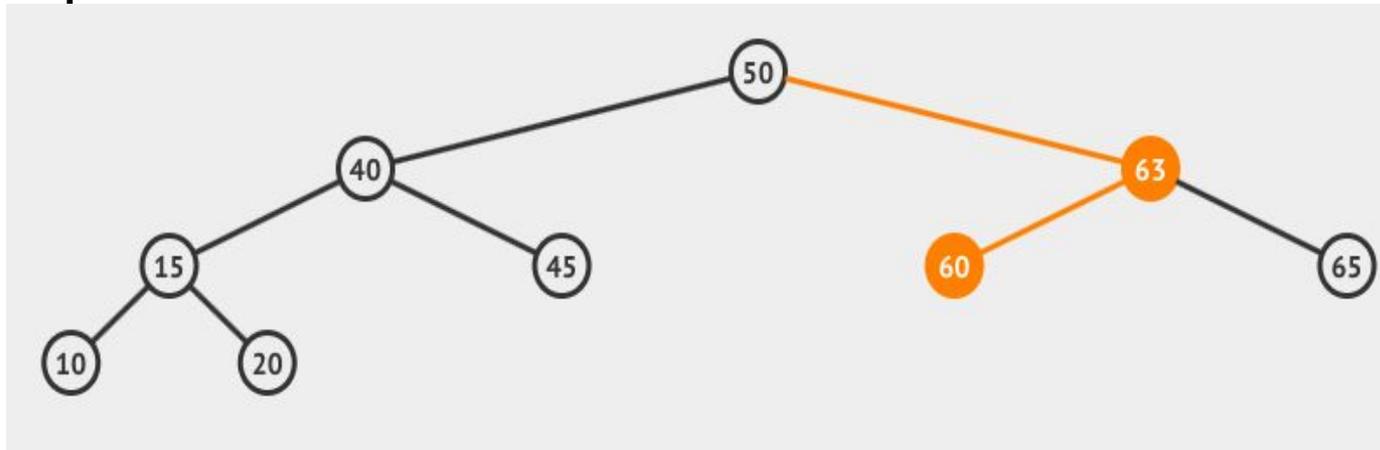
Primera rotación derecha: 63 se convierte en el hijo derecho de 60. Ahora ya podríamos aplicar una rotación simple izquierda sobre 60:



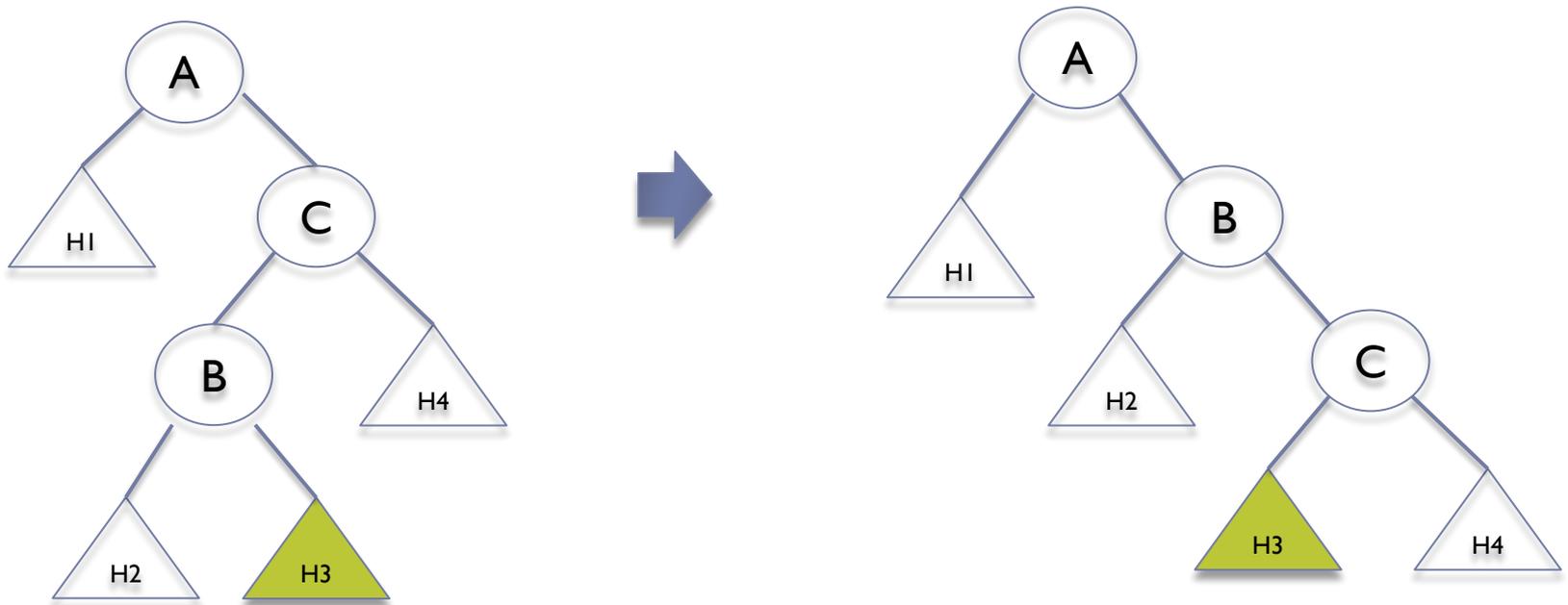
Doble Rotación Derecha Izquierda



Segunda rotación derecha: rotación simple izquierda sobre 60. En el árbol resultante, todos los nodos están equilibrados.

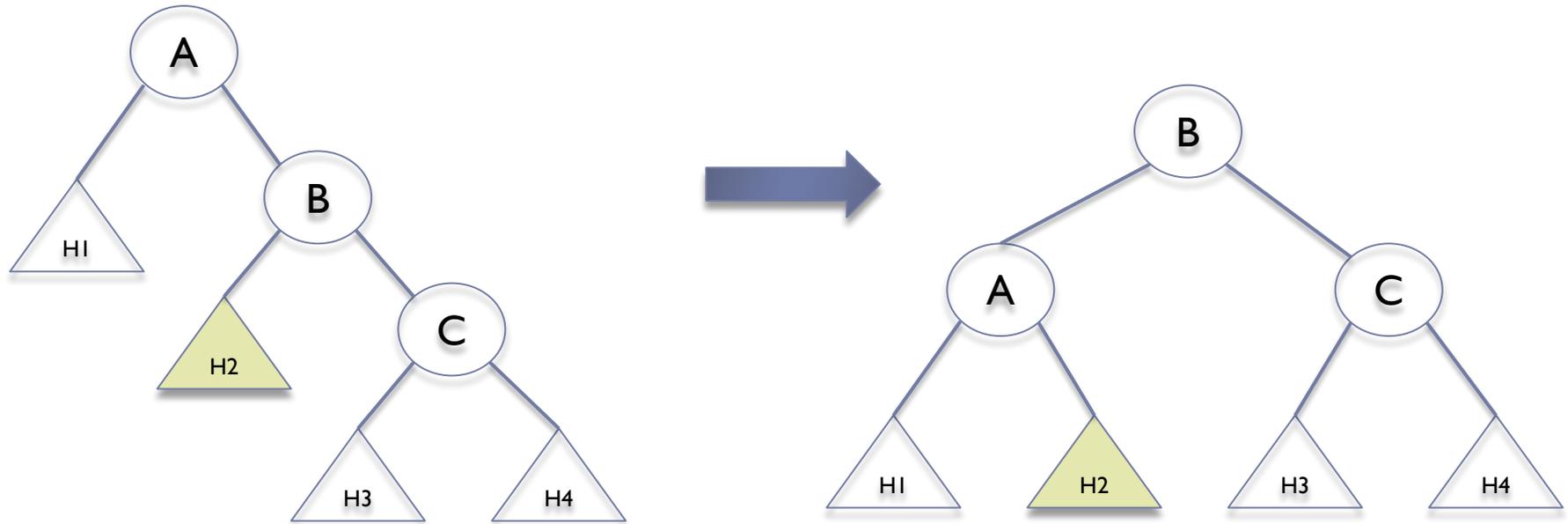


Doble Rotación Derecha Izquierda



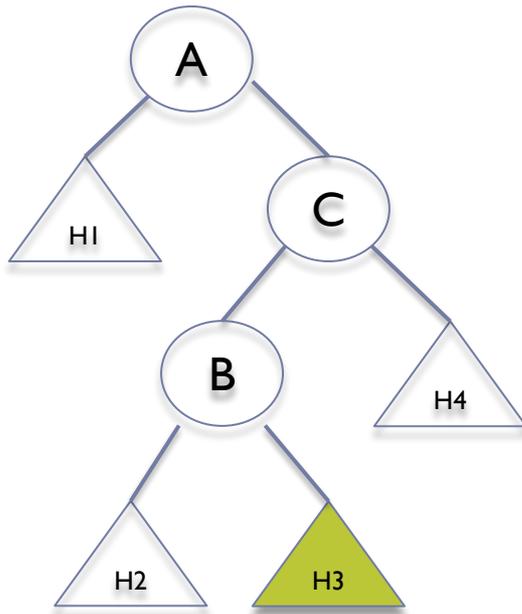
Primera rotación: B como hijo derecho de A y padre de C. El antiguo hijo derecho de B, debe pasar a ser hijo izquierdo de C

Doble Rotación Derecha Izquierda

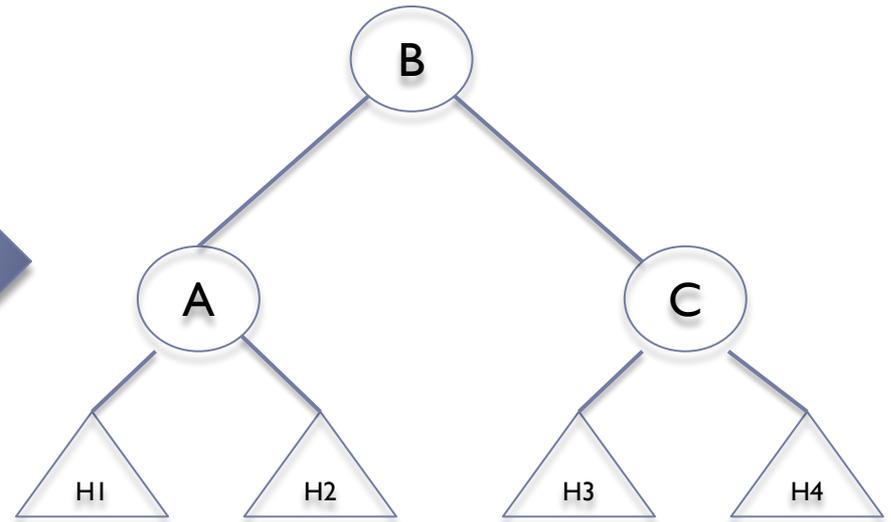


Segunda rotación: A gira a su izquierda y B se convierte en la nueva raíz. El antiguo hijo izquierdo de B debe pasar a ser hijo derecho de A, para que siga siendo un ABB.

Doble Rotación Derecha Izquierda



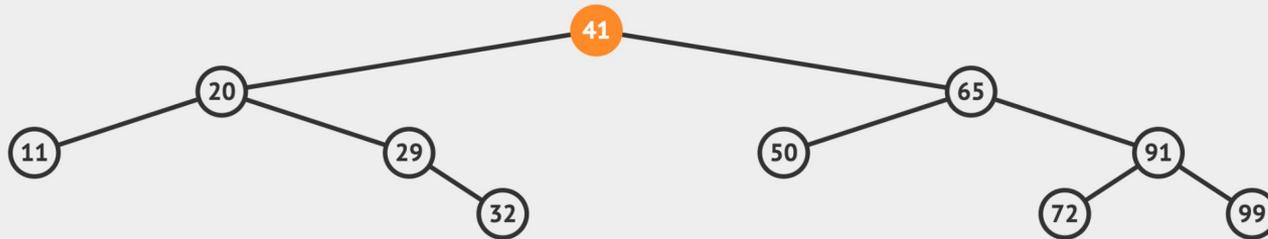
InOrder: $H_1 A H_2 B H_3 C H_4$



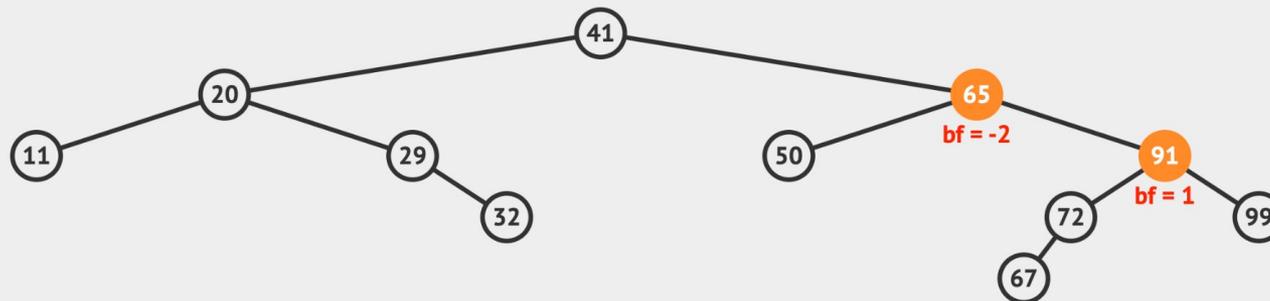
InOrder: $H_1 A H_2 B H_3 C H_4$

=

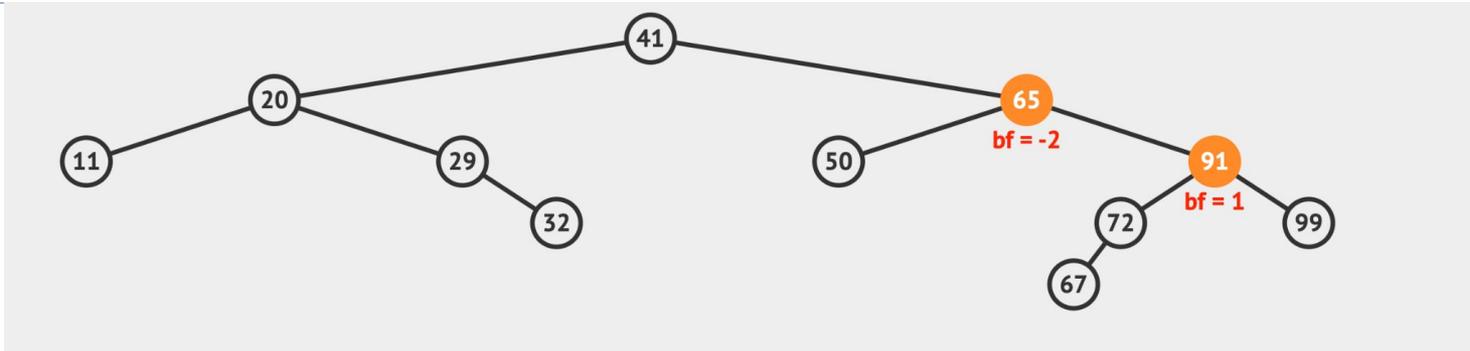
Doble Rotación Derecha Izquierda



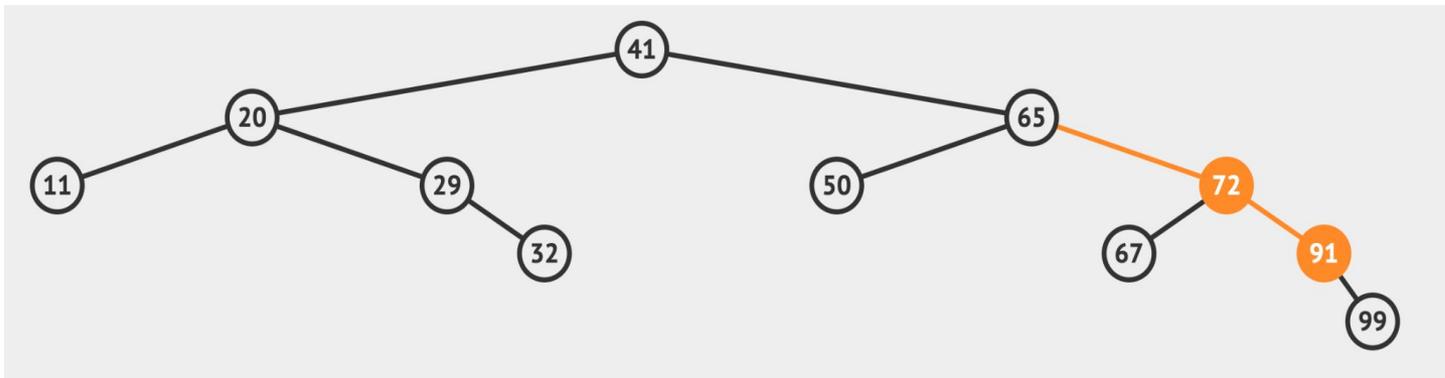
En el anterior árbol equilibrado, insertamos 67. Esto produce que el nodo 65 quede desbalanceado. Su rama más larga viene por la izquierda (91), y luego por la derecha (72). Debemos aplicar una doble rotación derecha,-izquierda



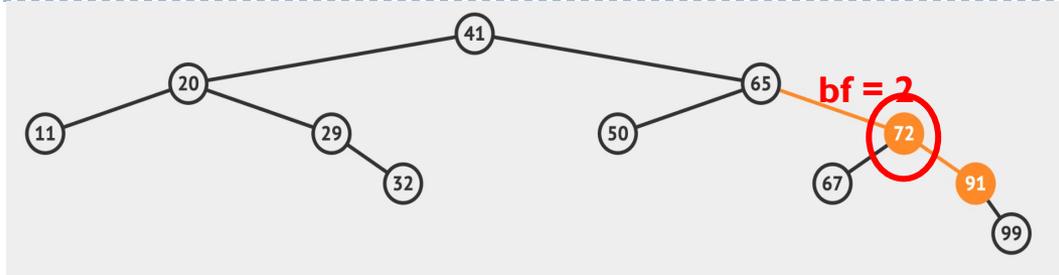
Doble Rotación Derecha Izquierda



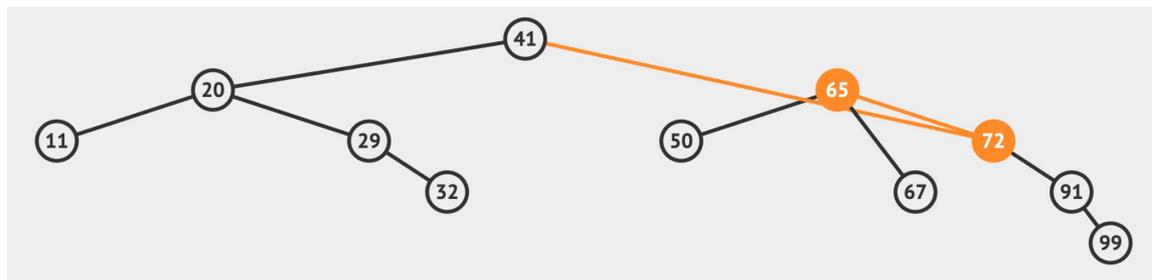
Primera rotación: simple derecha. El nodo 72 debe convertirse en el hijo derecho del nodo desequilibrado: 65. El nodo 65 sigue desequilibrado y ahora ya podemos aplicarle la rotación simple izquierda.



Doble Rotación Derecha Izquierda

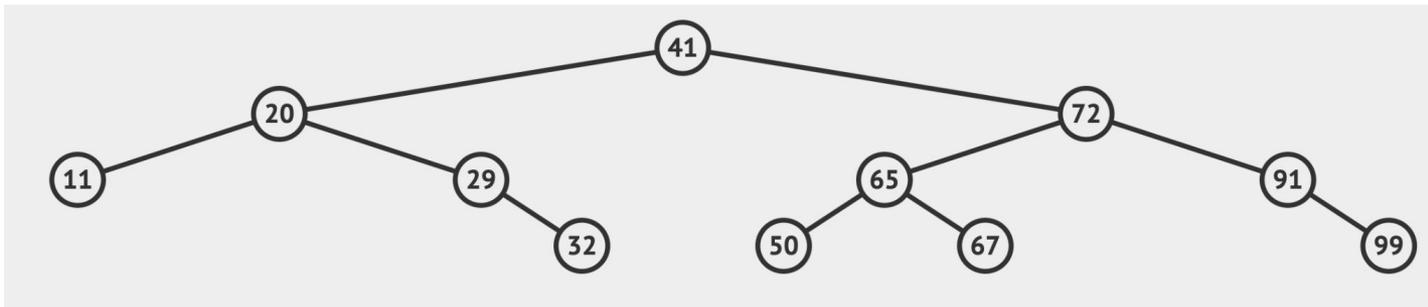


Segunda rotación: simple izquierda. El nodo con elemento 72 deberá convertirse en la nueva raíz del subárbol, y el nodo desbalanceado, nodo con elemento 65, deberá convertirse en su hijo izquierdo. Como el nodo 72 ya tiene hijo izquierdo, nodo con elemento 67, dicho nodo se convertirá en el hijo derecho del nodo con elemento 65.



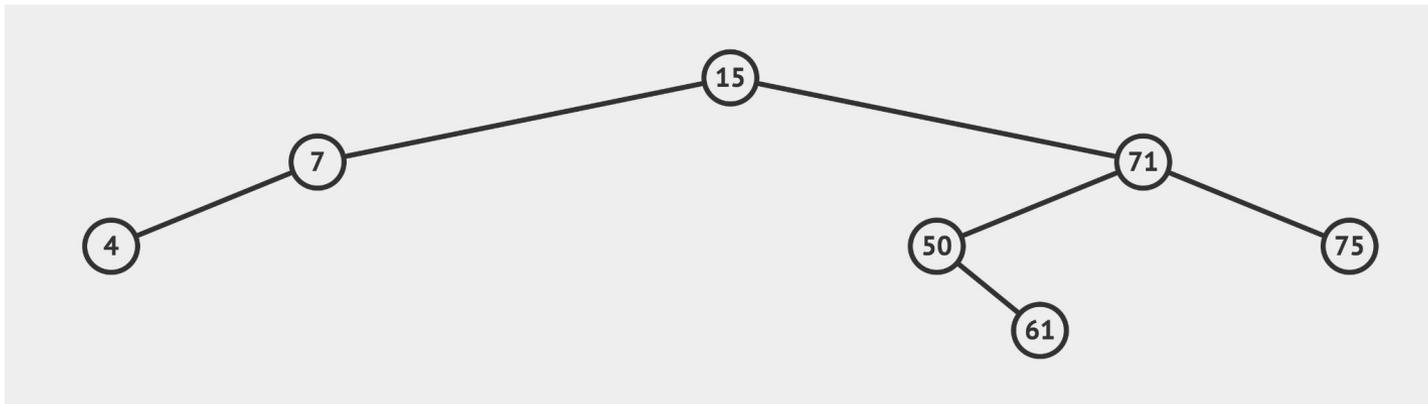
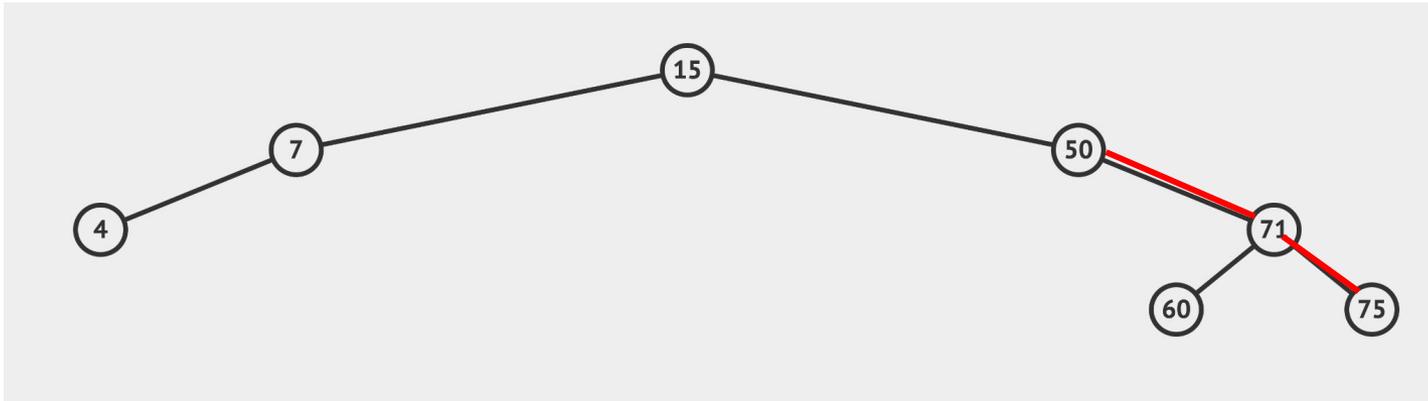
Doble Rotación Derecha Izquierda

Finalmente, en el árbol resultante, todos los nodos ya están equilibrados



Si se pueden aplicar dos rotaciones distintas

Elegir la rotación más simple.



¿Cuándo aplicar las rotaciones?

- Después de cada operación de inserción o borrado, se debe comprobar si el nodo ha quedado desequilibrado (solo en la rama donde se ha insertado o eliminado) y aplicar la rotación correspondiente.

Implementación

- Puedes encontrar la implementación de un árbol AVL en el siguiente [enlace](#).