

Árboles rojo negro

Clase 09

IIC 2133 - Sección 1

Prof. Sebastián Bugedo

Sumario

Obertura

Eliminación en ABB

Eliminación en Rojo-negros

Epílogo

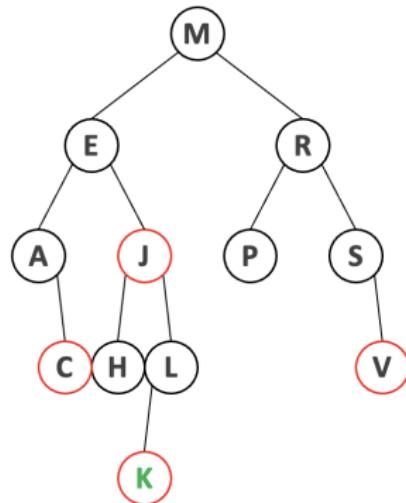
Árboles rojo-negro

Definición

Un **árbol rojo-negro** es un ABB que cumple

1. Cada nodo es rojo o negro
2. La raíz del árbol es negra
3. Si un nodo es rojo, sus hijos deben ser negros
4. La cantidad de nodos negros camino a cada hoja desde un nodo cualquiera debe ser la misma

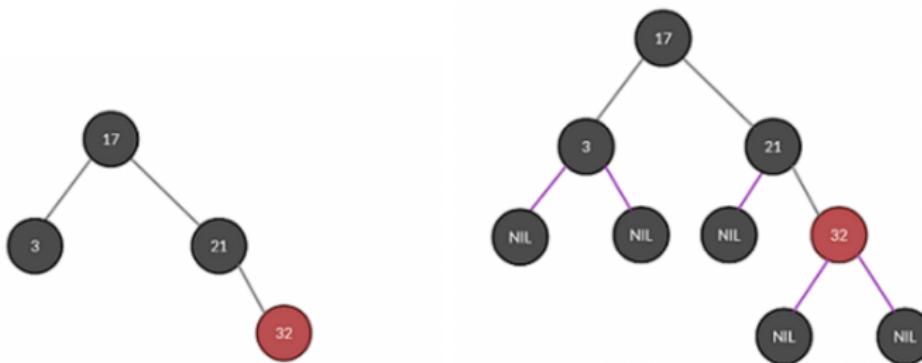
Adicionalmente, las hojas vacías se consideran nodos negros



Recordemos esta última propiedad y cómo ayuda al balance

Árboles rojo-negro

En el siguiente árbol, cada hijo vacío es un nodo negro



Considerar estos nodos vacíos ayuda a
asegurar que el árbol no sea *tan* desbalanceado

Árboles rojo-negro

Las inserciones en árboles rojo-negro involucran la inserción y un re-balanceo

- Si el padre del nodo insertado es negro: FIN
- Si el padre es rojo: análisis según el color del tío
 - Tío negro: rotaciones y cambios de color (local)
 - Tío rojo: cambios de color (recursivo)

La creación de un árbol con esta estrategia de inserción asegura balance

¿Qué pasa si queremos borrar nodos?

¿Cómo mantenemos el balance?

Objetivos de la clase

- Eliminar nodos en rojo-negros y balancear en distintos escenarios

PRIMER DATICONCURSO!



Link al concurso: DatiConcurso #1

Plazo de recepción de creaciones: 28 de septiembre 23:59

Sumario

Obertura

Eliminación en ABB

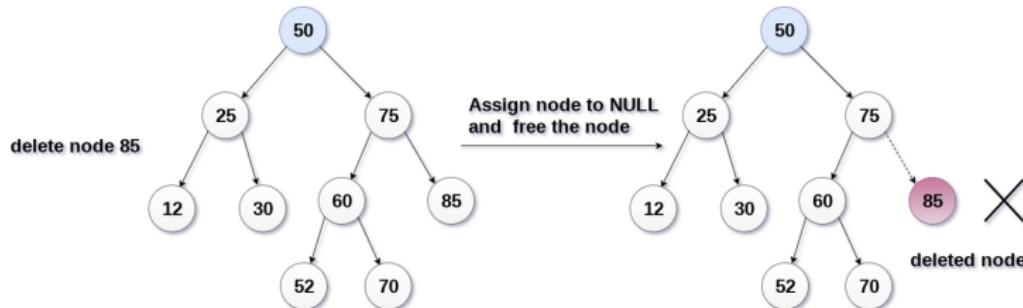
Eliminación en Rojo-negros

Epílogo

Eliminación en ABB

Al igual que en la inserción, estudiaremos diversos casos

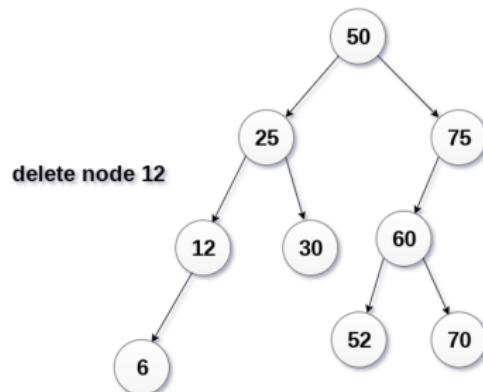
Pero antes, recordemos la eliminación en ABB



Caso hoja: simplemente se elimina

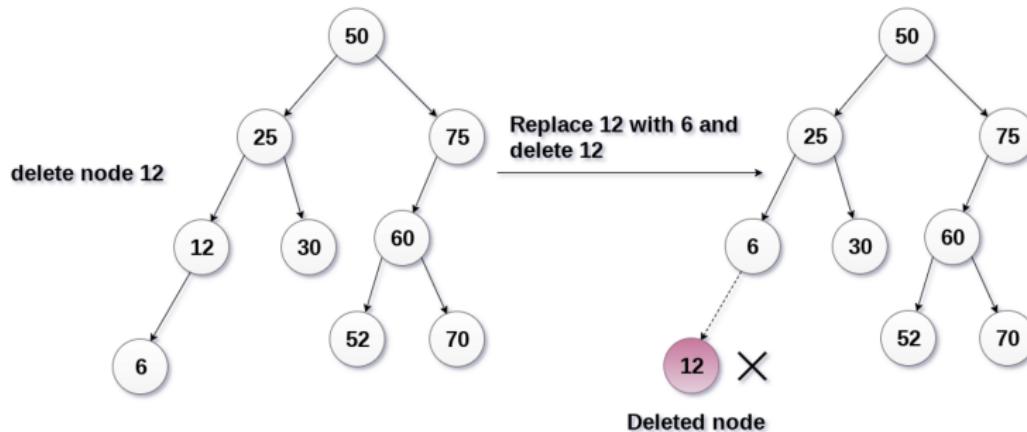
Eliminación en ABB

Al borrar padre con hijo único



Eliminación en ABB

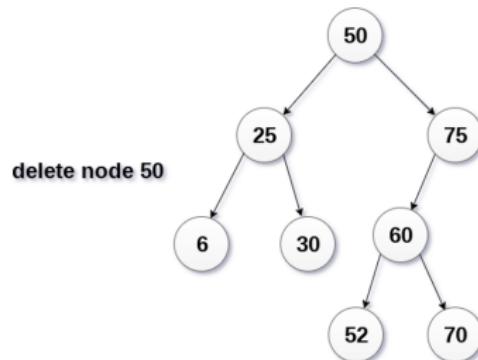
Al borrar padre con hijo único



Caso padre con 1 hijo: intercambio y se borra la hoja

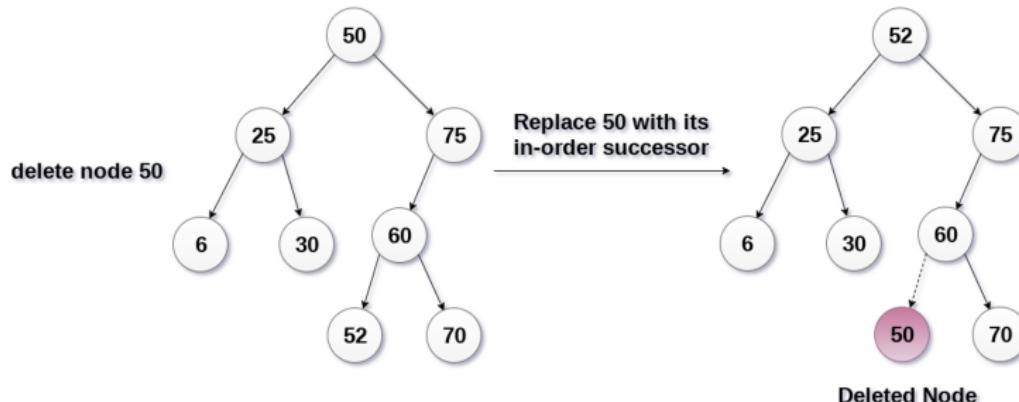
Eliminación en ABB

Al borrar padre con dos hijos



Eliminación en ABB

Al borrar padre con dos hijos



Caso padre con 2 hijos: intercambio con sucesor (o antecesor)
y se borra recursivamente esa antigua posición

Eliminación en ABB

Con ayuda de los métodos `min` y `max` podemos manejar los tres casos de **eliminación**

1. Si es hoja: se borra
2. Si es padre con 1 hijo: intercambio padre-hijo y se borra la hoja
3. Si es padre con 2 hijos: se intercambia con sucesor y se borra esa antigua posición
 - Si el sucesor no era hoja, se elimina recursivamente

¿Podemos utilizar esta misma estrategia para borrar en árboles rojo-negro?

Sumario

Obertura

Eliminación en ABB

Eliminación en Rojo-negros

Epílogo

Eliminación en árboles rojo-negro

La **eliminación** en árboles rojo-negro, al igual que la inserción, involucra dos fases

1. Eliminación como en un ABB cualquiera
2. Recuperación de propiedades rojo-negro

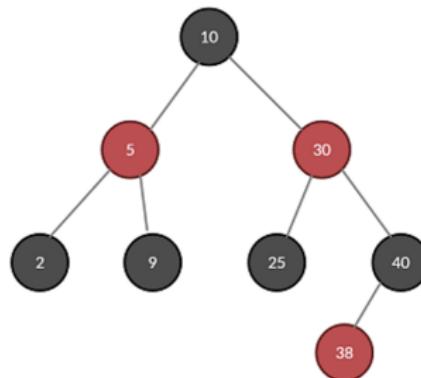
Veremos 5 escenarios de esta última fase

Escenario 1

Tal como en la eliminación en ABB, habrá un caso trivial

Caso 1

Si el nodo es hoja roja: simplemente se elimina



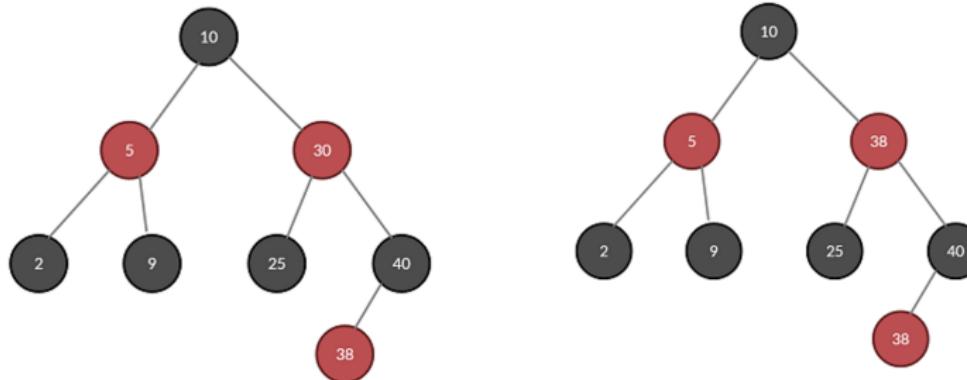
Ejercicio

Elimine el nodo 30 del árbol rojo-negro anterior



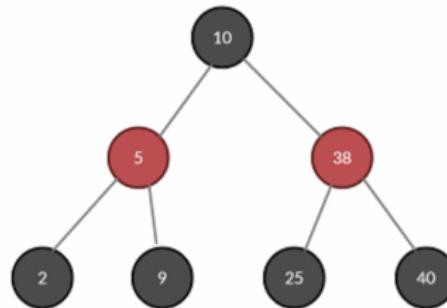
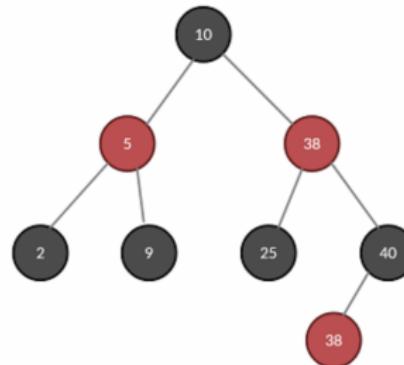
Escenario 1

Primero se intercambia con su sucesor como en árbol ABB cualquiera



Escenario 1

Luego se borra como hoja roja



Escenarios no triviales

El escenario 1 es el único en que simplemente borramos

Cuando el nodo borrado no sea hoja roja, lo reemplazaremos por un nuevo tipo de nodo **double black (DB)**

Si el **DB** era hoja, queda como **DB vacío**. En ciertos contextos nos encontraremos también con nodos **DB con valor**

Los siguientes escenarios nos muestran cómo manejar los **nodos DB** de manera **recursiva**

Escenario 2

Varios de los escenarios serán recursivos hacia la raíz, por lo que definimos un caso base

Caso 2

Si el **nodo DB** es raíz, entonces se cambia a **nodo negro**

Escenario 3

Caso 3

Si el **nodo DB** tiene hermano negro y los hijos de su hermano son negros

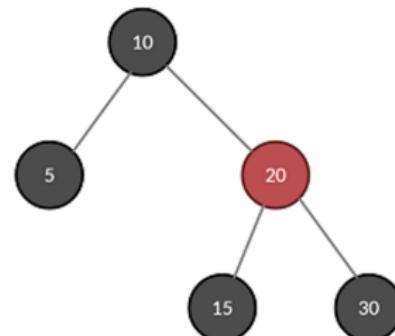
1. Si es DB vacío, se borra y todo nodo DB pasa a ser negro
2. Si es DB con valor, pasa a ser **negro**
3. Cambiar hermano del nodo DB a **rojo**
4. Según el color del padre del nodo DB
 - 4.1 Padre negro: volverlo **DB**
 - 4.2 Padre rojo: volverlo **negro**

Escenario 3

Caso 3

Si el **nodo DB** tiene hermano negro y los hijos de su hermano son negros

1. Si es DB vacío, se borra y todo nodo DB pasa a ser negro
2. Si es DB con valor, pasa a ser **negro**
3. Cambiar hermano del DB a **rojo**
4. Según el color del padre del DB
 - 4.1 Padre negro: volverlo **DB**
 - 4.2 Padre rojo: volverlo **negro**



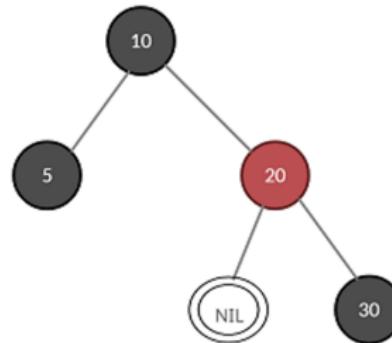
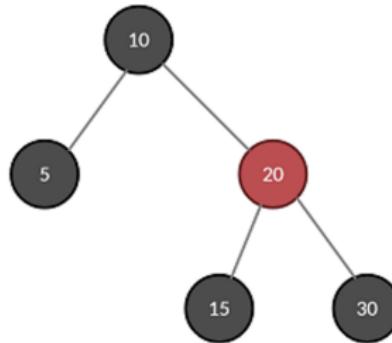
Ejercicio

Elimine el nodo 15 del árbol rojo-negro anterior



Escenario 3

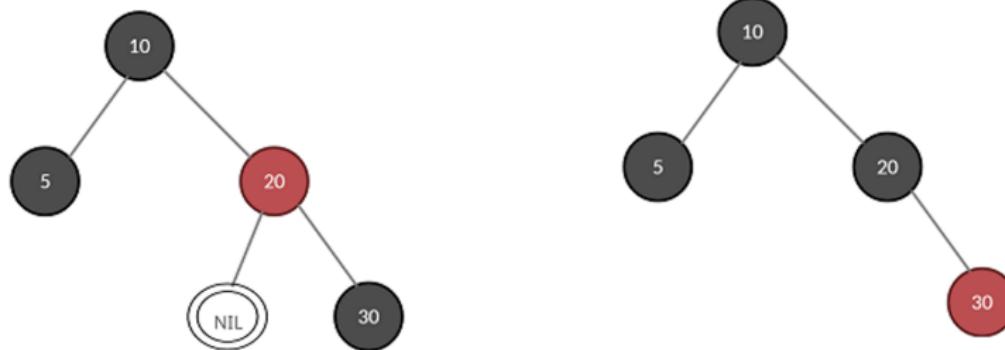
Primero se borra como en ABB, y como era negro queda como DB



Escenario 3

Luego, como su hermano es negro y los hijos de este son negros (E3)

- Se borra el DB por ser vacío
- Se cambia su hermano a rojo
- Se cambia su padre a negro

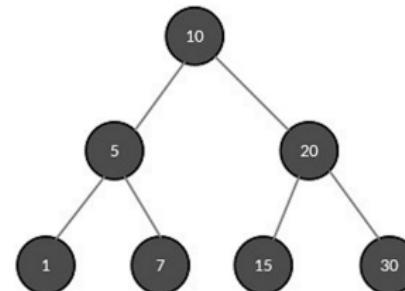


Escenario 3

Caso 3

Si el **nodo DB** tiene hermano negro y los hijos de su hermano son negros

1. Si es DB vacío, se borra y todo nodo DB pasa a ser negro
2. Si es DB con valor, pasa a ser **negro**
3. Cambiar hermano del DB a **rojo**
4. Según el color del padre del DB
 - 4.1 Padre negro: volverlo **DB**
 - 4.2 Padre rojo: volverlo **negro**



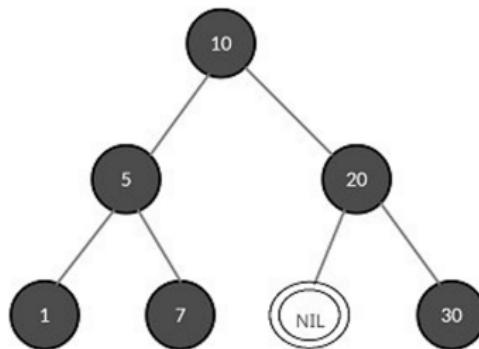
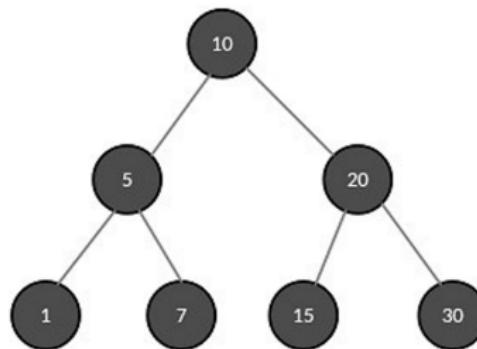
Ejercicio

Elimine el nodo 15 del árbol rojo-negro anterior



Escenario 3

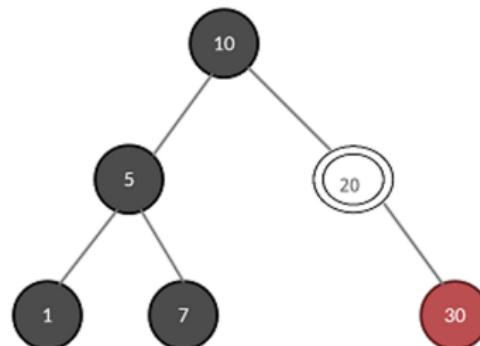
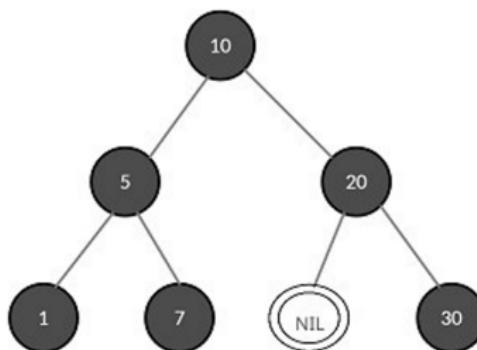
Primero se borra como en ABB, y como era negro queda como DB



Escenario 3

Luego, aplica el E3

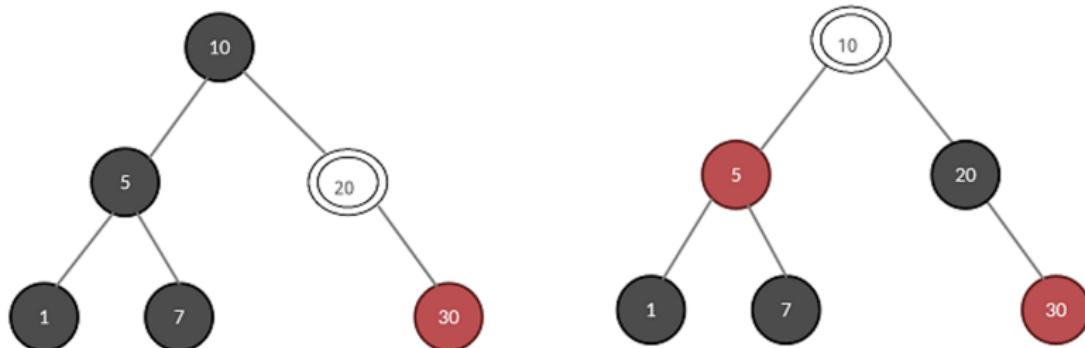
- Se borra el DB por ser vacío
- Se cambia su hermano a rojo
- Se cambia su padre a DB (porque ya era negro)



Escenario 3

Luego, nos centramos en el nuevo DB: aplica el E3 nuevamente

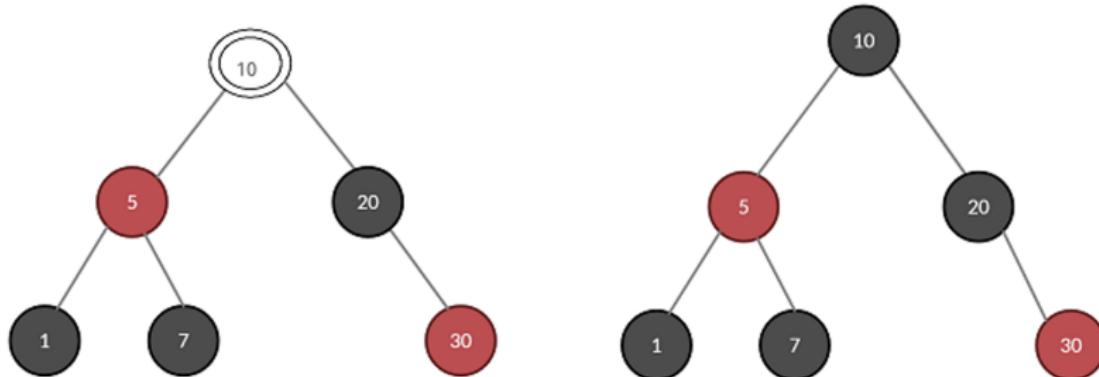
- Se cambia el DB a negro
- Se cambia su hermano a rojo
- Se cambia su padre a DB (porque ya era negro)



Escenario 3

Llegamos a la raíz y aplica el E2

- Se cambia el DB a negro por ser raíz



Escenario 4

Caso 4

Si el **nodo DB** tiene hermano rojo

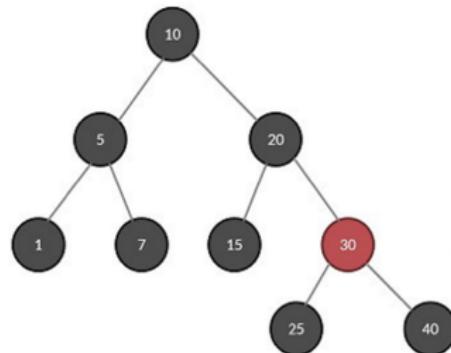
1. Intercambiar color del padre y hermano
2. Rotación padre-hermano en dirección al DB
3. Verificar qué escenario aplica a este nuevo árbol

Escenario 4

Caso 4

Si el **nodo DB** tiene hermano rojo

1. Intercambiar color del padre y hermano
2. Rotación padre-hermano en dirección al DB
3. Verificar qué escenario aplica a este nuevo árbol



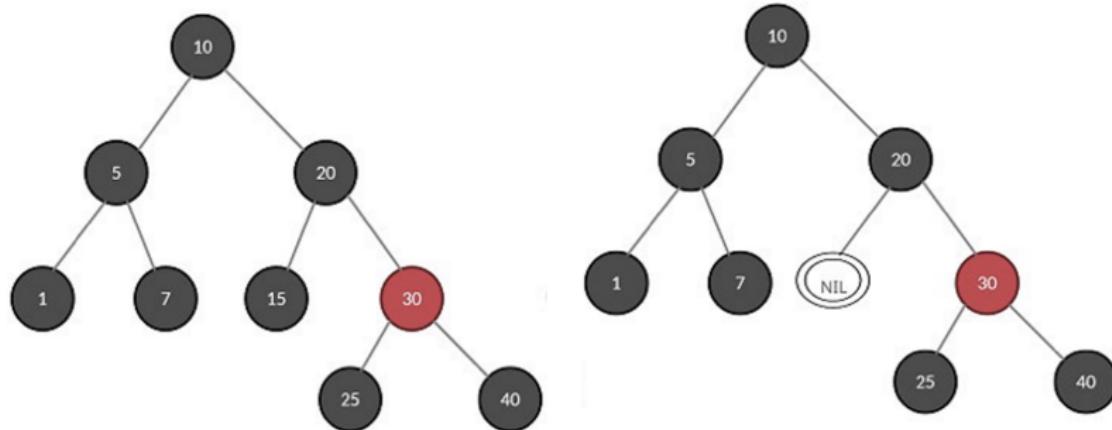
Ejercicio

Elimine el nodo 15 del árbol rojo-negro anterior



Escenario 4

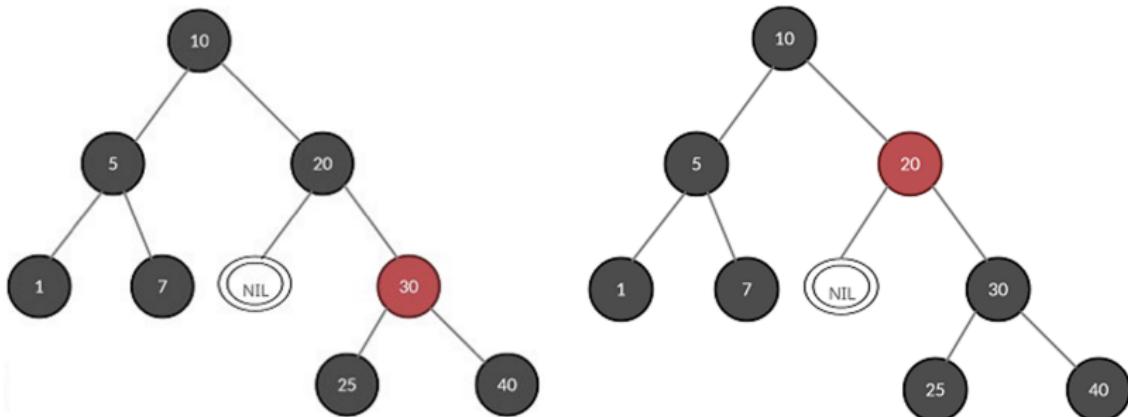
Primero se borra como en ABB, y queda como DB vacío



Escenario 4

Aplica el E4 por tener hermano rojo

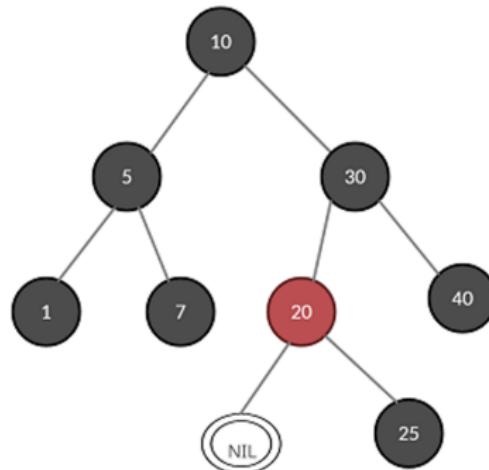
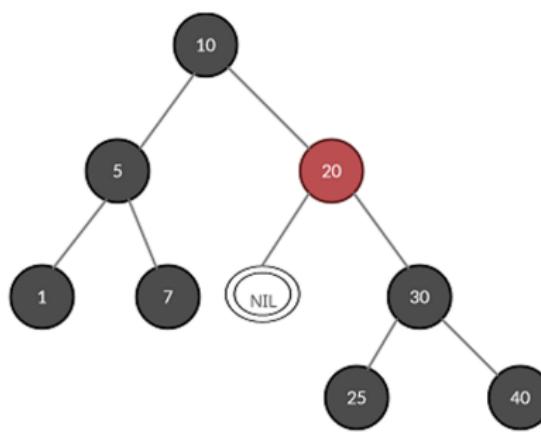
- Se intercambia el color de padre y hermano



Escenario 4

Aplica el E4 por tener hermano rojo

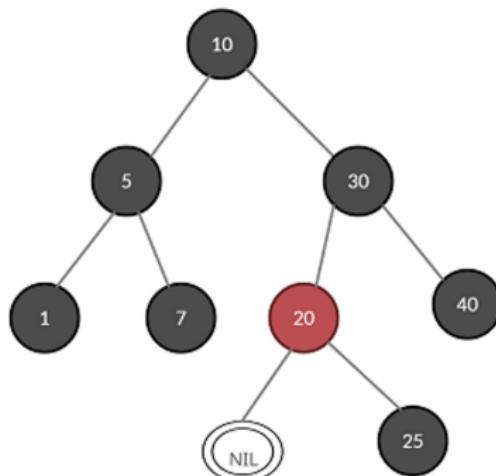
- Se rota padre-hermano en dirección al DB



Escenario 4

Aplica el E4 por tener hermano rojo

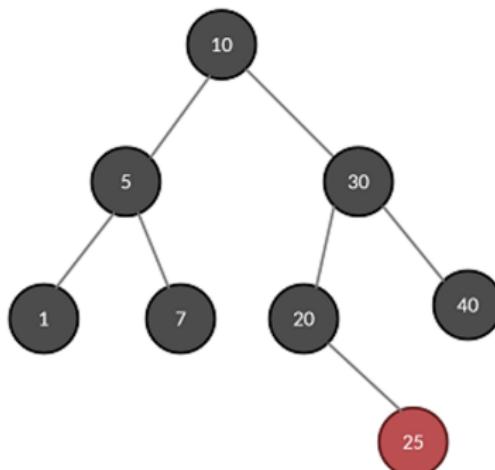
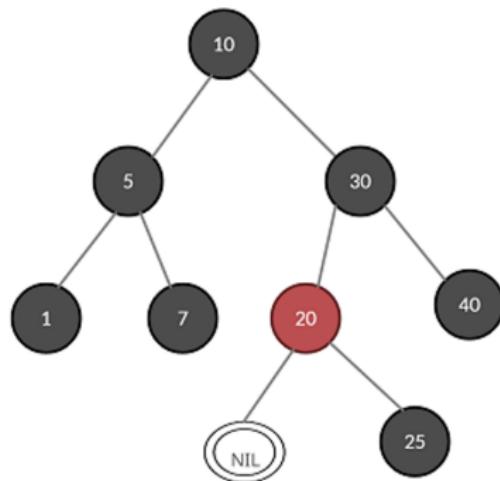
- Se verifica qué escenario aplica: E3 (hermano negro con hijos negros)



Escenario 4

Se aplica el E3

- Se borra el DB por ser vacío
- Se cambia su hermano a rojo
- Se cambia su padre a negro



Escenario 5

Caso 5

Si el **nodo DB** tiene hermano negro, sobrino lejano negro y sobrino cercano rojo

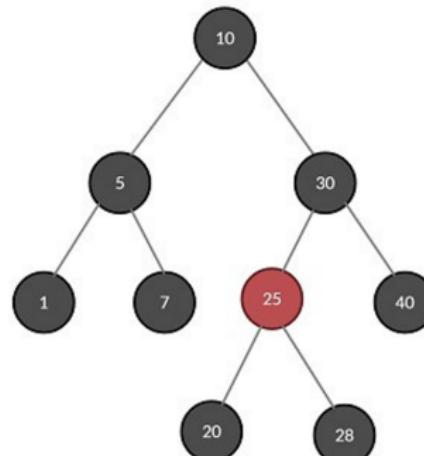
1. Intercambiar color del hermano con sobrino rojo
2. Rotación hermano-sobrino en dirección opuesta al DB
3. Aplicar E6

Escenario 6

Caso 6

Si el **nodo DB** tiene hermano negro y sobrino lejano rojo

1. Intercambiar color del padre con hermano
2. Rotación padre-hermano en dirección al DB
3. Pasar DB a negro
4. Pasar ex sobrino lejano rojo a negro



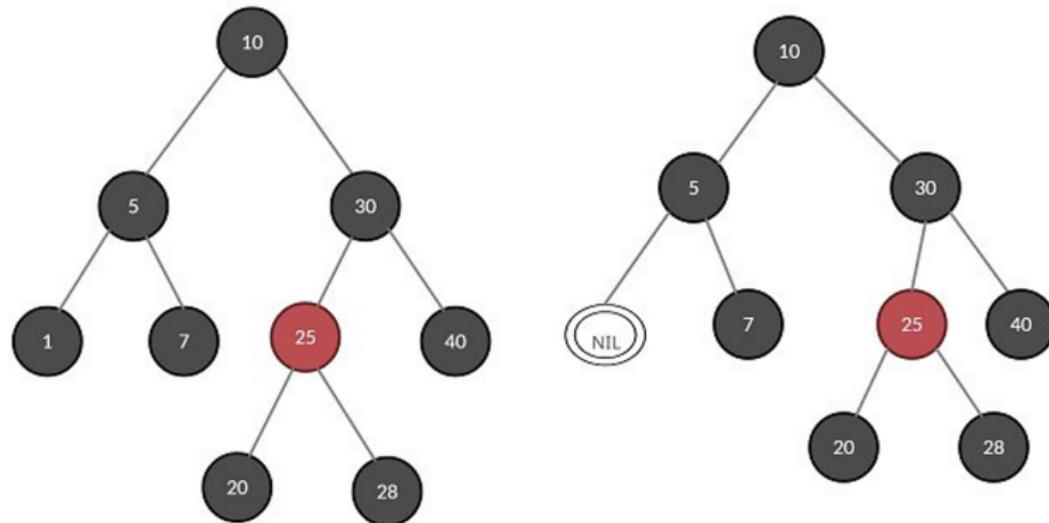
Ejercicio

Elimine el nodo 1 del árbol rojo-negro anterior



Escenario 5-6

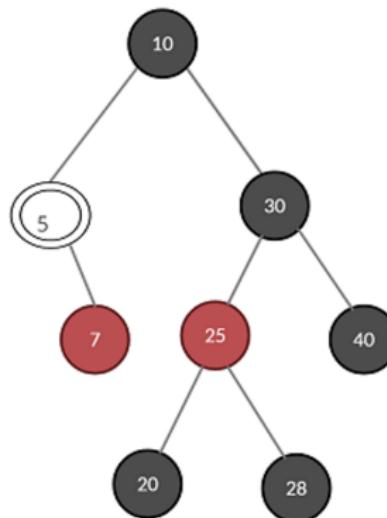
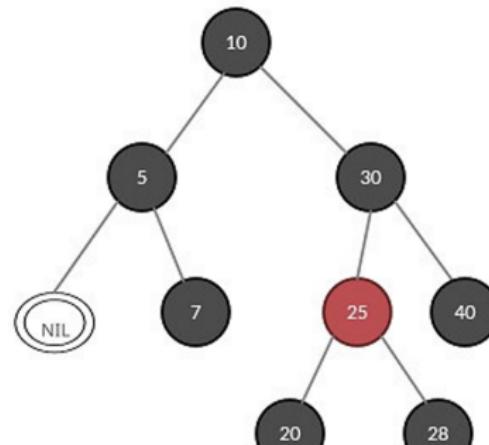
Primero se borra como en ABB, y queda como DB vacío



Escenario 5-6

Ahora aplica el caso E3 (hermano negro con hijos negros)

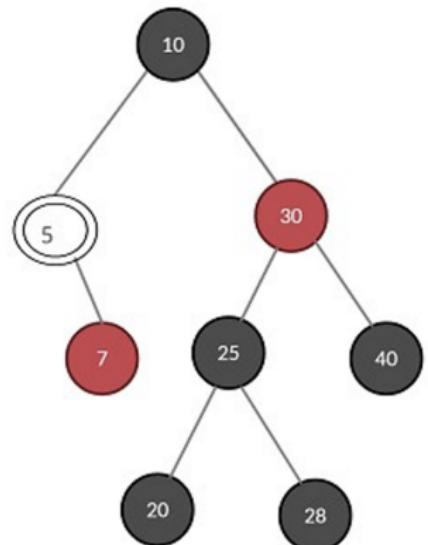
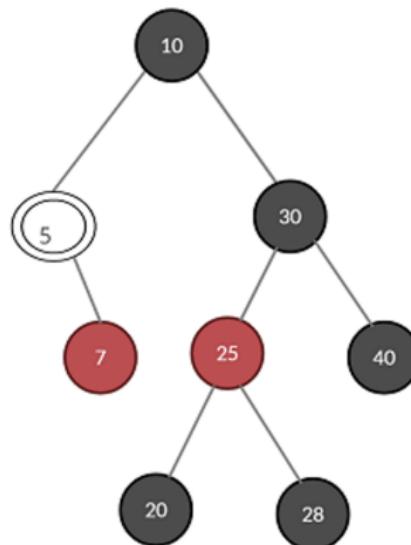
- Se borra el DB por ser vacío
- Se cambia su hermano a negro
- Se cambia su padre a DB (ya era negro)



Escenario 5-6

Ahora aplica el caso E5 (hermano negro, sobrino cercano rojo, sobrino lejano negro)

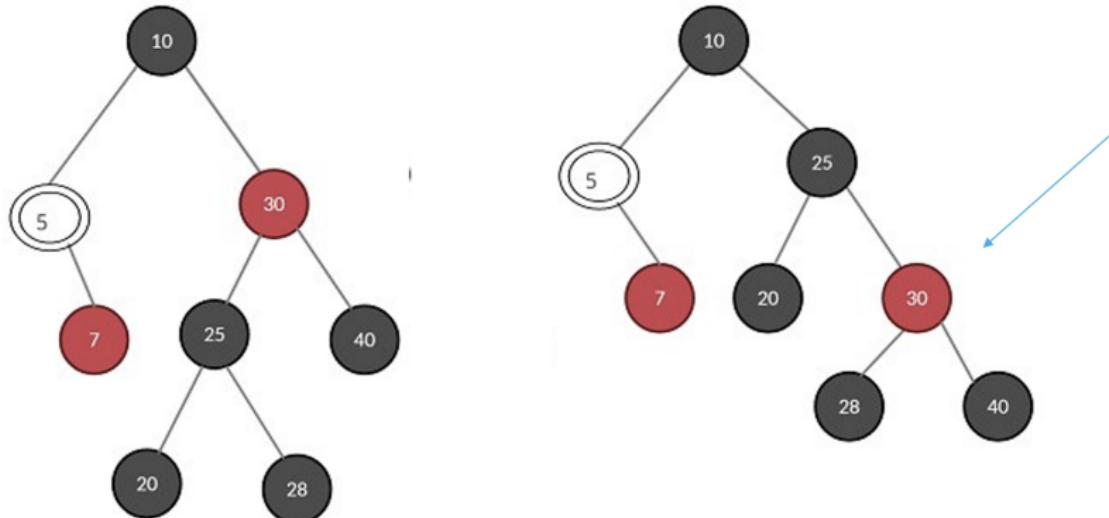
- Intercambiar colo hermano - sobrino rojo



Escenario 5-6

Ahora aplica el caso E5 (hermano negro, sobrino cercano rojo, sobrino lejano negro)

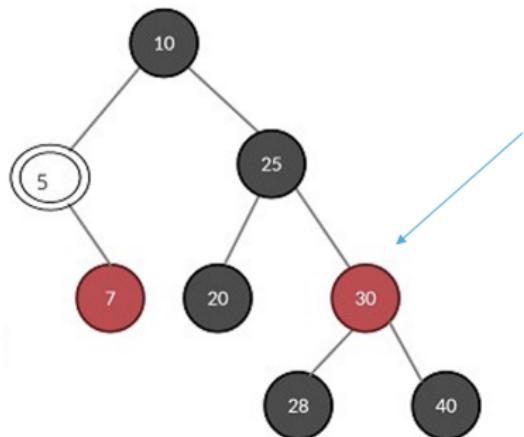
- Rotación hermano-sobrino en dirección opuesta al DB



Escenario 5-6

Ahora aplica el caso E5 (hermano negro, sobrino cercano rojo, sobrino lejano negro)

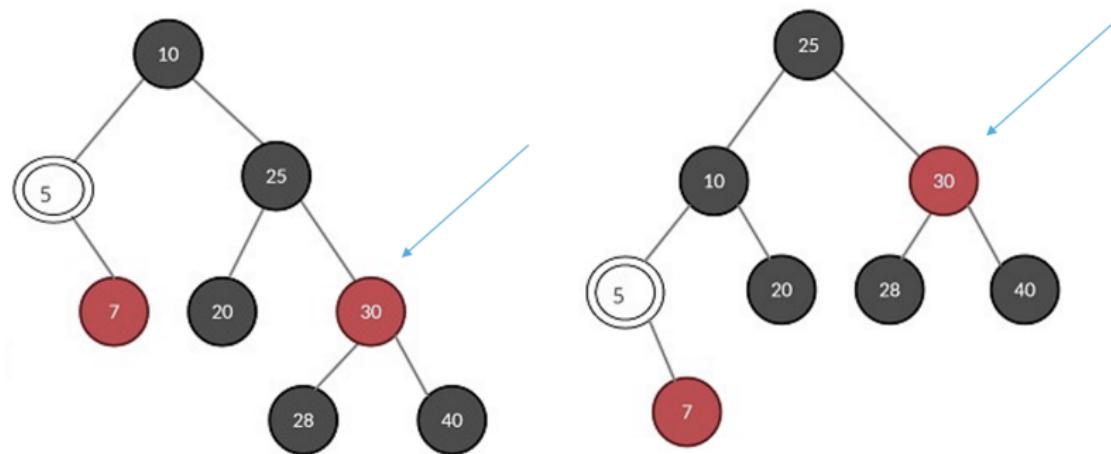
- Aplicar E6 (hermano negro, sobrino lejano rojo)



Escenario 5-6

Ahora aplica el caso E6 (hermano negro, sobrino lejano rojo)

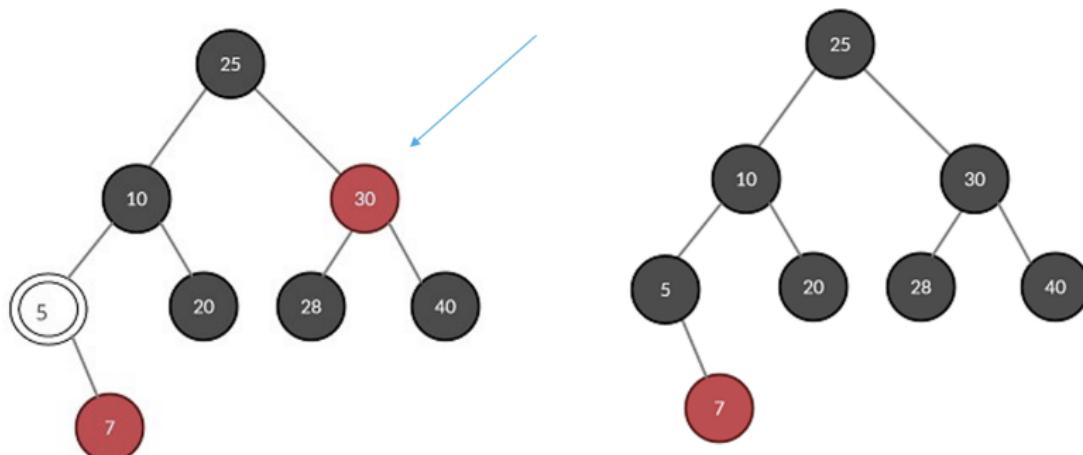
- Intercambiar color del padre con hermano
- Rotación padre-hermano en dirección al DB



Escenario 5-6

Ahora aplica el caso E6 (hermano negro, sobrino lejano rojo)

- Pasar DB a negro
- Pasar ex sobrino lejano rojo a negro



Sumario

Obertura

Eliminación en ABB

Eliminación en Rojo-negros

Epílogo

Epílogo

Vea

www.menti.com

Introduce el código

8698 3695



O usa el código QR