

Auxiliar 7 - Árboles Balanceados

Profesores: Nelson Baloian
Jeremy Barbay
Patricio Poblete

Auxiliares: Gabriel Flores, Sven Reisenegger
Juglar Díaz, Gabriel Norambuena
Cristóbal Muñoz

P1. Árbol Rojo-Negro

Un árbol Rojo-Negro es un tipo de árbol de búsqueda binaria auto balanceante. Esto quiere decir que intenta a toda costa mantener mínima su altura y evitar 'ramas' muy largas. Cumple las siguientes propiedades:

- (a) La raíz es negra.
- (b) Todo nodo rojo debe tener dos nodos hijos negros.
- (c) Cada camino desde un nodo dado a sus hojas descendientes contiene el mismo número de nodos negros.

- **Caso 1:** *El nodo insertado es la raíz.*

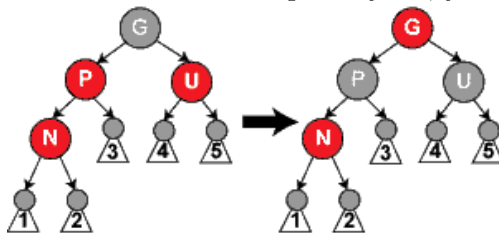
La raíz se pinta de negro.

- **Caso 2:** *El padre del nodo insertado es negro.*

Se deja igual.

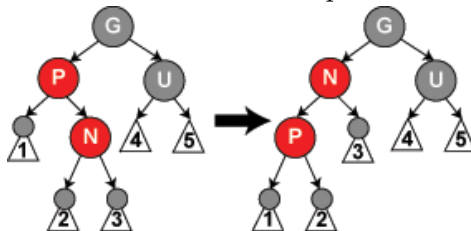
- **Caso 3:** *El padre del nodo y el tío son rojos.*

Se cambia de color al padre y tío, y se revisa el caso 1 para el abuelo.

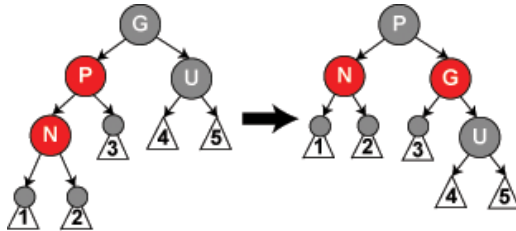


- **Caso 4:** *El padre es rojo pero el tío es negro. El nodo insertado está hacia adentro.*

Se hace una rotación simple sobre el padre para poder pasar al caso 5.



- **Caso 5:** *El padre es rojo pero el tío es negro. El nodo insertado está hacia afuera.*
 Se hace una rotación simple sobre el abuelo, y se hacen cambios de color.



Dibuje paso a paso el árbol Rojo-Negro resultante luego de insertar cada uno los siguientes elementos: $\{2, 1, 4, 5, 9, 3, 6, 7\}$, indique además todos los estados intermedios.

P2. Árbol-B

El árbol-B es otro tipo de árbol auto balanceante ordenado donde cada nodo puede tener hasta M hijos y $M-1$ claves, esto se llama un árbol-B de orden M . Las propiedades de este árbol son las siguientes:

- Cada nodo tiene como **máximo** M hijos.
- Cada nodo (excepto la raíz) tiene como **mínimo** $M/2$ hijos.
- La raíz tiene al menos 2 hijos si no es una hoja.
- Todos los nodos hojas están a la misma altura.
- Un nodo con K hijos tiene $K-1$ claves almacenadas.
- Para todo nodo se cumple que su k -ésimo hijo tiene valores mayores que la clave $k-1$ y menores que la clave k . Si una clave no existe (en los bordes) se ignora esa condición. Por ejemplo, el primer hijo cumple que tiene valores menores que la primera clave.

El proceso de inserción parte igual que la inserción en un ABB, primero se realiza una búsqueda de la clave y se encuentra el nodo hoja en el cuál debería insertarse. Aquí hay 2 opciones:

- Si la hoja no está llena, se inserta el valor y se termina de manera normal.
- Si la hoja está llena, se toma el valor central C entre todas su claves y se "sube" un nivel. El resto de los valores es dividido en 2 nodos, uno con todas las claves menores que C y otro con todas las claves mayores. Si el nodo que se dividió era la raíz, se crea una nueva raíz con una sola clave y 2 hijos.

Para comprobar la utilidad de este árbol (y entender mejor la inserción) compararemos la inserción de un mismo set de claves en un ABB y en un árbol-B de orden 5. Las claves a insertar son los números del 1 al 25, de manera ascendente (peor caso ABB).