Respuesta I2 P4

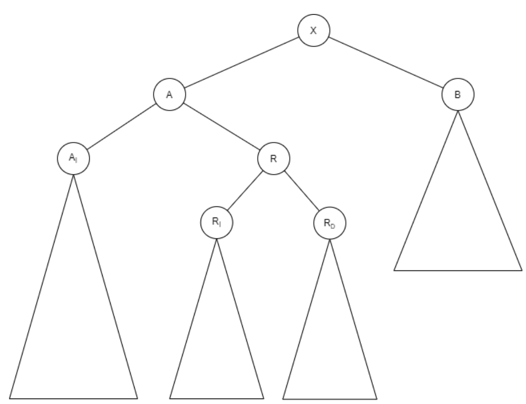
a) De clases se sabe que un árbol AVL está balanceado cuando el módulo de la diferencia entre las alturas del hijo izquierdo, y el derecho, de la raíz no son mayores a 1, y además cada hijo está balanceado. Por esto, si tenemos dos árboles A y B de igual altura, y una llave x que es mayor a todas las llaves de A y menor que todas las llaves de B, entonces solo basta colocar a la llave x como raíz del nuevo árbol C, colocar a la raíz de A como hijo izquierdo de x, y a la raíz del árbol B como hijo derecho de x.

Esto funciona, ya que, como A y B tienen la misma altura, y el añadir un padre a ambos árboles aumenta en 1 la altura de ambos, luego si la altura anterior de A y B era *N*, la nueva altura de ambos será *N + 1*, y como la diferencia de alturas es 0, el árbol C está balanceado.

Es $O(1)$ porque solo se debe agregar a los hijos de la llave x a las raíces de A y B.

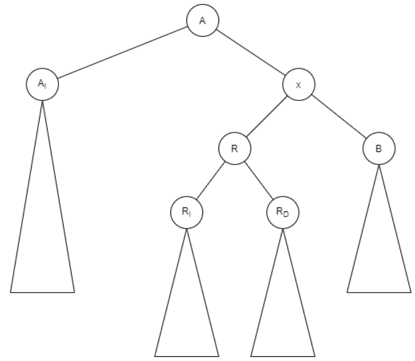
b) Si las alturas difieren en uno, se puede utilizar el mismo mecanismo del punto a). Definiendo a la altura como H. $H(A) = N$, $H(B) = N \pm 1$. Luego, igual que antes, el añadir un padre a cada uno aumenta en uno la altura. $H(A*) = N + 1$, $H(B*) = N \pm 1+1$, y el balance de C se define como $B(C) = H(A) - H(B) = N+1-(N\pm1+1) = \pm1$. Luego, como el módulo del balance es menor o igual a 1, se cumple la condición de balance. Es $O(1)$ por la razón del punto a).

c) En este caso, hay que cambiar el método. Se explicará el caso en el que la altura de A sea mayor que la de B, pero el caso opuesto es análogo. Realizamos el mismo método anterior para obtener un árbol C. Luego, se toma el hijo derecho de A, al cual llamaremos R.



Luego, nos encontramos con dos casos:

1. $H(A_I) \geq H(R)$: En este caso, basta con realizar una rotación derecha entre A y x, quedando:



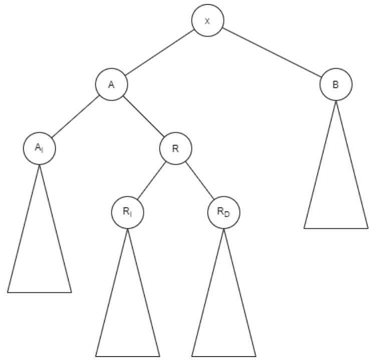
Como R es mayor que el nodo A y menor que x, es correcto hacer la rotación, manteniendo la propiedad de orden binaria. Y es balanceado porque $H(A_I) \geq H(R)$ y $H(B) = H(A) - 2$, luego, la rotación disminuirá en 1 la altura del lado izquierdo, quedando $H(A) - 1$, y aumentará en 1 o 2 la altura del lado derecho, dependiendo de la altura de R, quedando o $[H(A)-2]+1 = H(A)-1$ o $[H(A)-2]+2 = H(A)$. En ambos casos, la diferencia con el lado izquierdo es menor o igual a 1, por lo que está balanceado. Además, $H(R) = H(A) - 1$, o $H(R) = H(A) - 2$, por lo que x también está balanceado (calculando la diferencia de $H(R)$ con $H(B)$).

2. $H(A_I) < H(R)$:

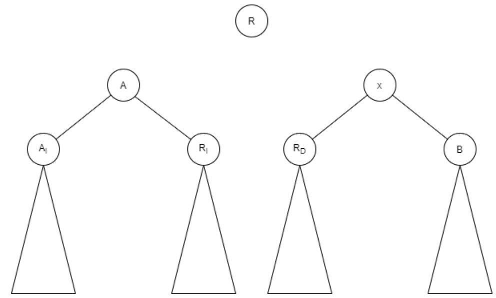
   a) En este caso, el método anterior no sirve, ya que entrega un árbol desbalanceado, por lo que hay que realizar otro método.

   b) El hijo izquierdo de R se deja como hijo derecho de la raíz de A, el hijo derecho de R se deja como hijo izquierdo de x.

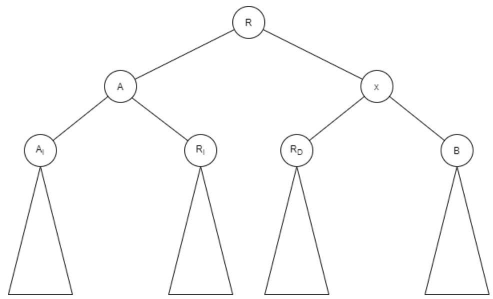
   c) Finalmente, se coloca a R como la raíz del árbol, dejando a la raíz de A como hijo izquierdo de R, y a la raíz de x como el hijo derecho de R.



(a)



(b)



(c)

Claramente, el nodo A es menor que R, ya que R era el hijo derecho de A. Luego, R_I es mayor que A, por propiedades del AVL. Además, R y R_D son menor que X, por enunciado. Entonces todas las posiciones son correctas.

Para las alturas, sabemos que $H(R) = H(A) - 1$, $H(A_I) = H(A) - 2$ y $H(B) = H(A) - 2$. Luego, todo queda balanceado, ya que $H(A_I) = H(B)$, y además $H(R_I)$ y $H(R_D)$ son o $H(A) - 2$, o $H(A) - 3$, manteniendo la diferencia máxima de 1 con $H(A_I)$ y $H(B)$ respectivamente.

d) Para responder esto nos apoyamos de las respuestas anteriores, ya que observamos que el caso propuesto se resume en que, partiendo desde la raíz del árbol agregado, se tiene que el hermano puede tener una altura menor en 1 o menor en 2, si la diferencia es de 1, el padre de esa raíz estará balanceado (por el punto b)), si no, habrá una diferencia de 2 por lo que, según el punto c), se puede balancear en O(1). Cabe destacar que tanto la raíz del nodo agregado, como su hermano, representarán árboles AVL con un padre en común que es siempre menor al primero pero mayor al segundo, coincidiendo con las condiciones del enunciado. Luego, se observa que ocurre lo mismo con el padre anterior, y se presentará el mismo problema con cada padre de la rama hasta la raíz. Como una rama de AVL tiene tamaño $\log n$, hay un orden de $\log n$ raíces a arreglar, y como cada raíz se arregla en $O(1)$, balancear el árbol tomará $O(\log n)$.

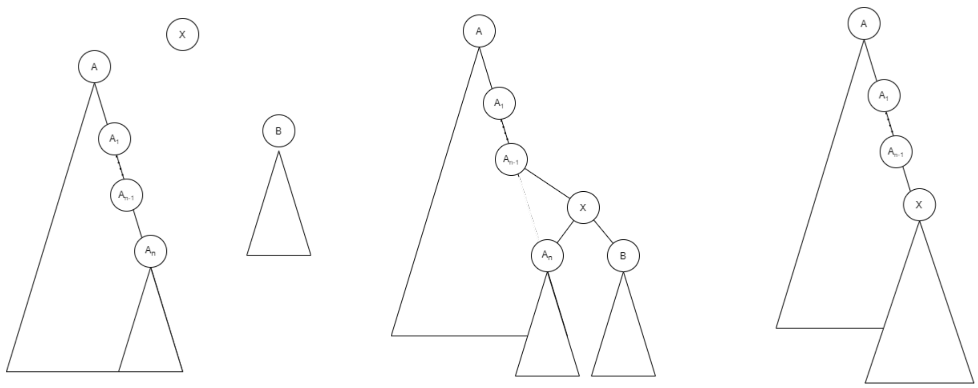
e) Se puede observar que hay tres casos posibles, en los que los tres siguen la misma lógica. Si A y B tienen la misma altura, basta con unirlos como se explica en a). Luego puede ser que A sea mayor que B, o el contrario. Se explicará el caso en que A tiene mayor altura a B, la demostración contraria es análoga.

Se debe buscar el primer nodo en A de la cadena de hijos derechos en tener una altura igual a la altura de B, luego se toma este nodo (al cual llamaremos A_n) y se coloca como hijo izquierdo de x, la raíz de B se coloca como el hijo derecho de x, y x se coloca en la posición inicial de A_n. Por a), el árbol en x será un AVL balanceado. Con este mecanismo se obtendrá una situación igual a la del punto d), ya que la altura del punto inicial de A_n creció en uno. Y como se demostró en d), este árbol es ordenable en $O(\log n)$. Se muestra un diagrama del proceso:

a) Se tienen el nodo x, y los árboles A y B, donde en A se marca la hilera de hijos derechos hasta que se encuentre el nodo A_n con altura igual a la altura de B.

b) Se coloca a A_n como hijo izquierdo de x, y B como hijo derecho de este mismo. Luego, x se coloca reemplazando al nodo en A:

c) Por último, se observa que se tiene la misma situación que en el punto d), donde se agrega un nuevo árbol aumentando la altura anterior en 1, y se sabe que esto es balanceable en $O(\log n)$ por lo explicado en el mismo punto.



(a) (b) (c)