

Segunda Parte:

Partiendo de un AVL/Árbol B, según sea el caso, se solicita realizar las operaciones explicitadas. Debe fundamentarse cada paso (incluyendo el nombre de la operación que interviene, si existiese) y realizarse el diagrama paso a paso.

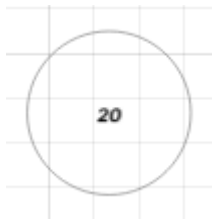
16. Partiendo de un AVL vacío, se solicita realizar las siguientes operaciones:

- A. Agregar el 20**
- B. Agregar el 29**
- C. Agregar el 21**
- D. Agregar el 12**
- E. Agregar el 7**
- F. Eliminar el 20**

Secuencia: +20, +29, +21, +12, +7, -20

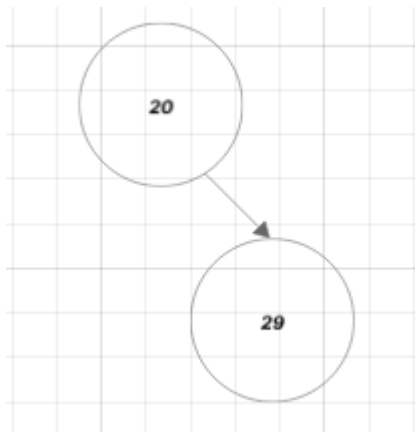
Usamos: $FB(n) = \text{altura(izq)} - \text{altura(der)}$ + rotaciones: LL ($-->$ Der), RR ($-->$ izq), LR, RL

A. Insertar 20 Árbol inicial y FB



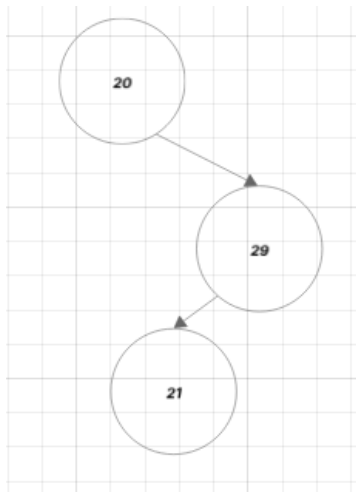
$FB(20)=0$

B. Insertar el 29: como $29 > 20$, va a la derecha

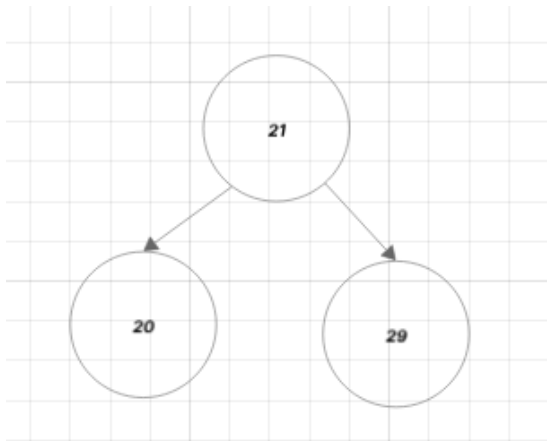


$FB(29)=0$, $FB(20) = -1$

C. Insertar 21: como $21 > 20$, va al lado derecho y como $21 < 29$, se ubica a la izquierda

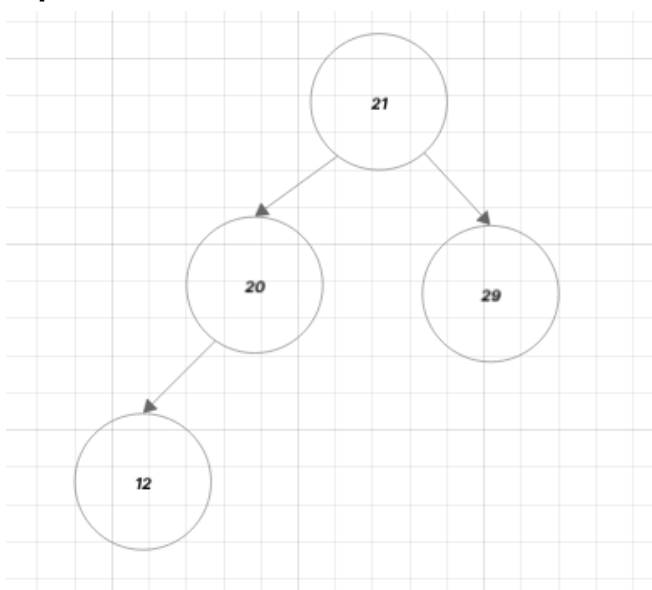


FB(21)=0, FB(29)=+1, FB(20)= -2 , desbalance en 20 (caso RL).
 Rotación DERECHA en 29 y rotación IZQUIERDA en 20:



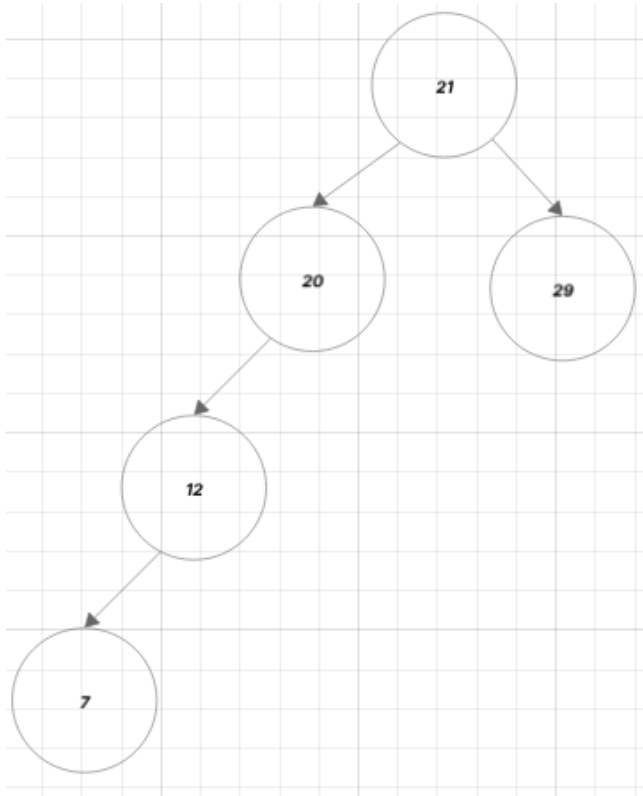
FB(20)=0, FB(29)=0, FB(21)=0

D. Insertar 12: como $12 < 21$, va al lado izquierda y como $12 < 20$ se ubica a la izquierda

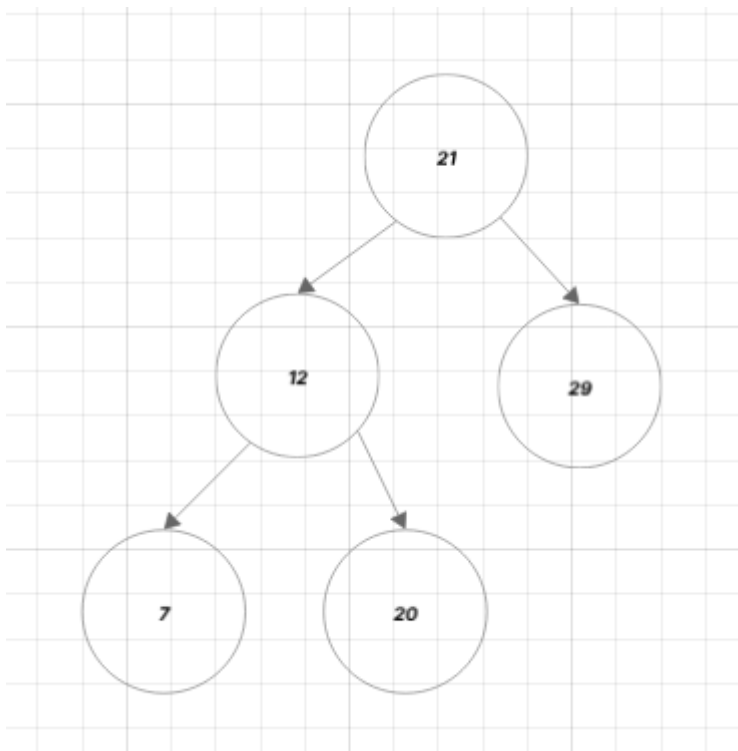


FB(12)=0, FB(20)=+1, FB(21)=+1, FB(29)=0

E. Insertar 7, como $7 < 21$ va al lado izquierda y a su vez como $7 < 20$ va nuevamente a la izquierda y como $7 < 12$ se ubica a la izquierda

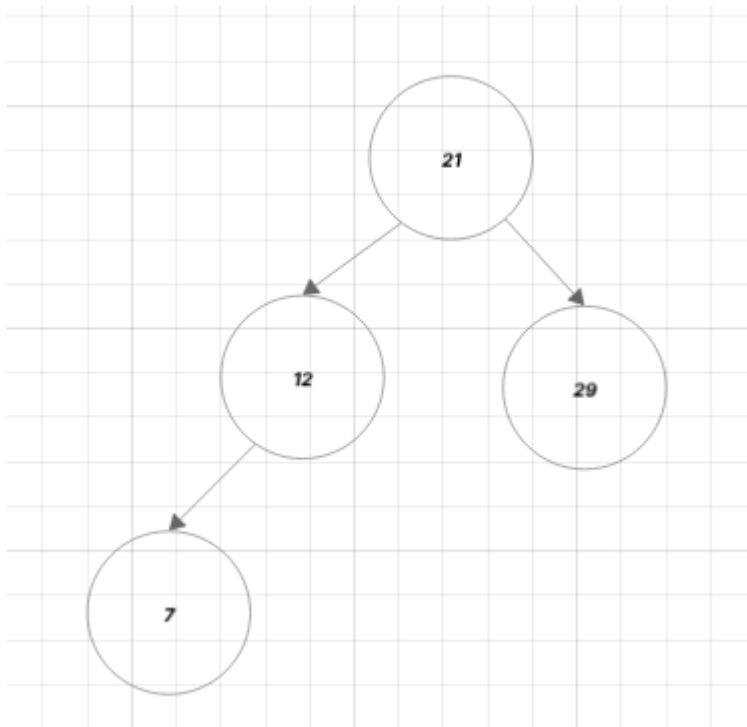


$FB(7)=0$, $FB(12)=+1$, $FB(20)=+2$, desbalance en 20 (caso LL)
 Rotación DERECHA sobre 20:



$FB(7)=0$, $FB(20)=0$, $FB(12)=0$, $FB(29)=0$, $FB(21)=1$

F. Eliminar 20: como 20 es hoja derecha de 12, se elimina directo quedando:



$FB(7)=0$, $FB(12)=1$, $FB(29)=0$, $FB(21)=+1$

Como Todos los $FB \in \{-1,0,+1\}$, el árbol está balanceado.

Rotaciones aplicadas:

(c) Caso RL en 20, (rotación derecha en 29 + rotación izquierda en 20)

(e) Caso LL en 20,(rotación derecha en 20)

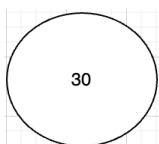
17. Partiendo de un AVL vacío, se solicita realizar las siguientes operaciones:

- Agregar el 30
- Agregar el 36
- Agregar el 10
- Agregar el 15
- Agregar el 12
- Eliminar el 30

Secuencia: +30, +36, +10, +15, +12, -30

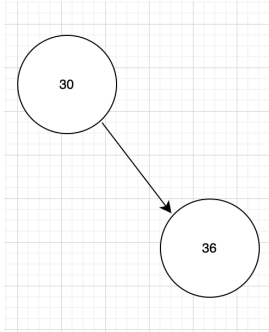
Usamos: $FB(n) = altura(izq) - altura(der)$ + rotaciones: LL (--> Der), RR (--> izq), LR, RL

a) Insertar 30 Árbol inicial y FB



- $FB(30)=0$

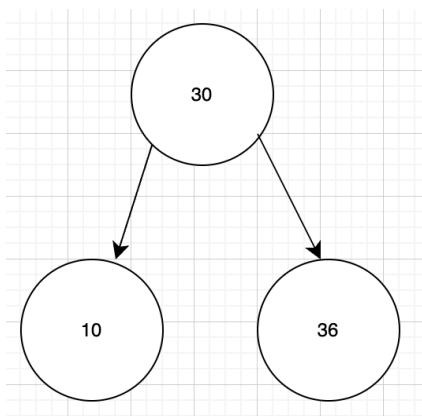
b) *Insertar 36*: 36 va a la derecha de 30



- $FB(36)=0$, $FB(30)=-1$

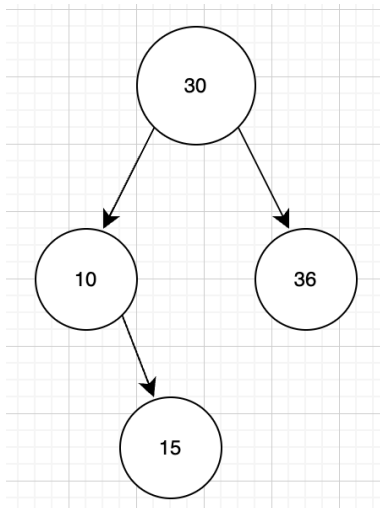
c) *Insertar 10*

10 va a la izquierda de 30



- $FB(10)=0$, $FB(36)=0$, $FB(30)=0$

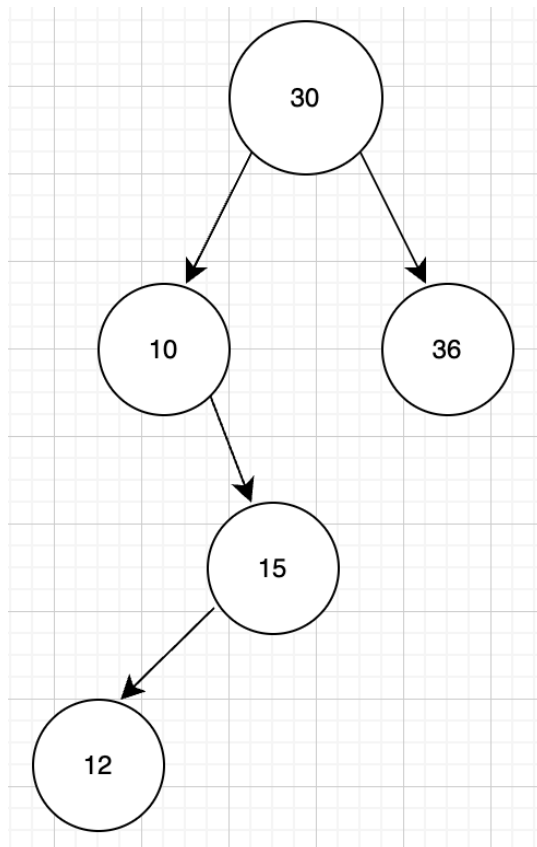
d) *Insertar 15*



• $FB(15)=0$, $FB(10)=-1$, $FB(36)=0$, $FB(30)=+1$

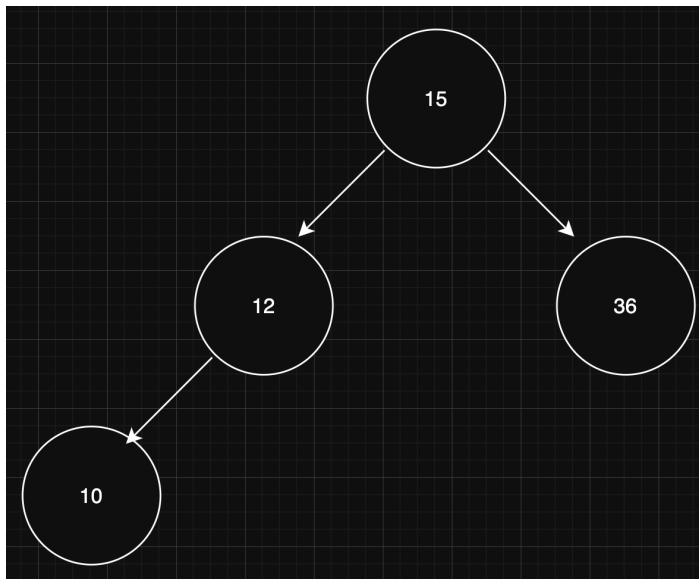
e) *Insertar 12*

12 cae en izq de 30 → der de 10 → izq de 15. Desbalance en 10 (caso RL).



• Caso RL en 10:

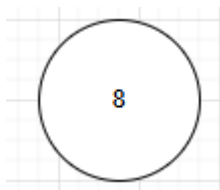
1) Rotación DERECHA en hijo (15)



18. Partiendo de un AVL vacío, se solicita realizar las siguientes operaciones:

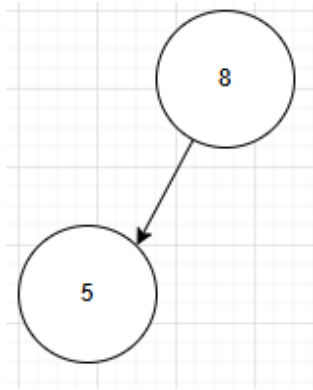
- a. Agregar el 8**
- b. Agregar el 5**
- c. Agregar el 6**
- d. Agregar el 3**
- e. Agregar el 1**
- f. Eliminar el 3**

a. Insertar 8



$$FB(8) = 0$$

b. Insertar 5



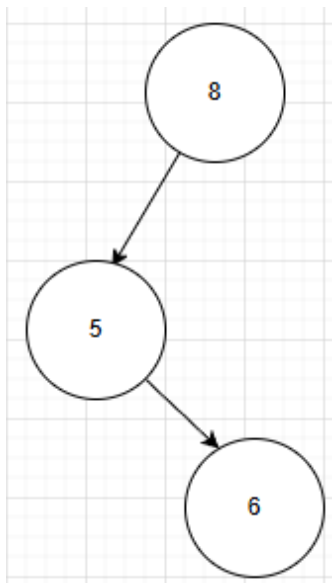
$FB(5)=0$, $FB(8)=+1$

No hay desbalance.

c. Insertar 6

$6 < 8 \rightarrow$ izquierda,

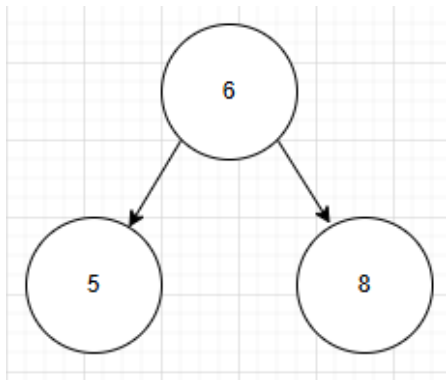
$6 > 5 \rightarrow$ derecha de 5



$FB(6)=0$, $FB(5) = -1$, $FB(8) = +2 \rightarrow$ desbalance en 8 \rightarrow caso LR

Rotaciones:

1. Rotación **izquierda** en 5
2. Rotación **derecha** en 8

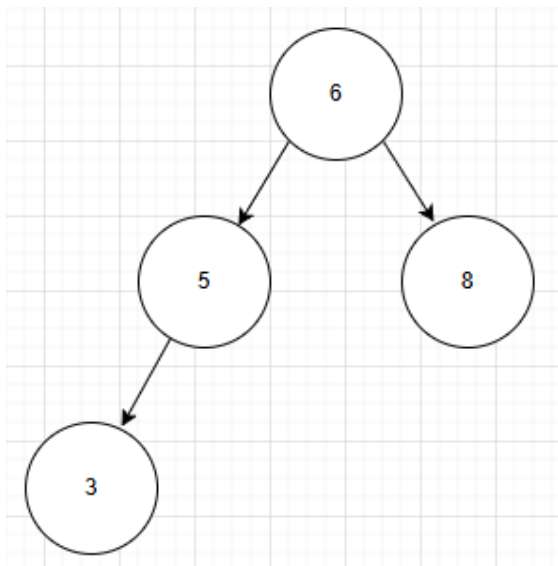


$FB(5)=0$, $FB(8)=0$, $FB(6)=0$

d. Insertar 3

$3 < 6 \rightarrow$ izquierda,

$3 < 5 \rightarrow$ izquierda de 5



$FB(3)=0$, $FB(5)=+1$, $FB(6)=+1$

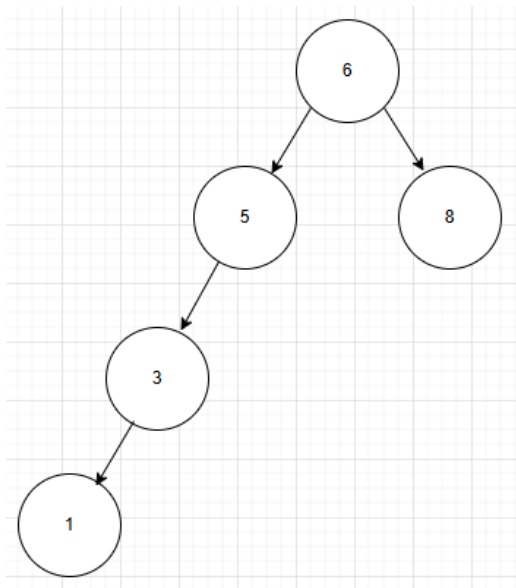
Todo balanceado.

e. Insertar 1

$1 < 6 \rightarrow$ izquierda,

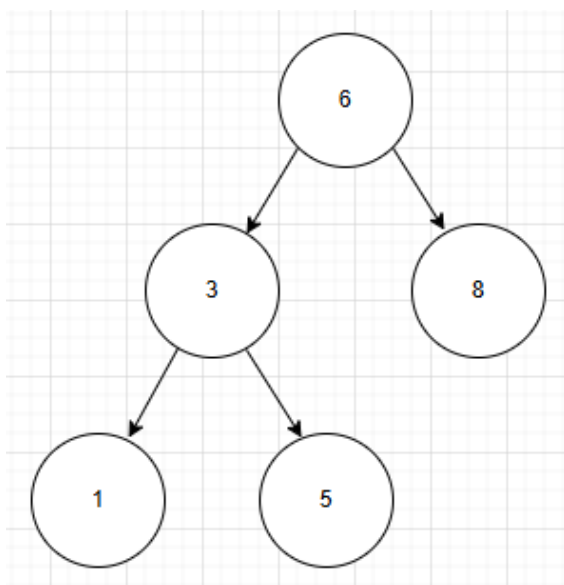
$1 < 5 \rightarrow$ izquierda,

$1 < 3 \rightarrow$ izquierda.



$FB(1)=0$, $FB(3)=+1$, $FB(5)=+2 \rightarrow$ desbalance en 5 (caso LL)

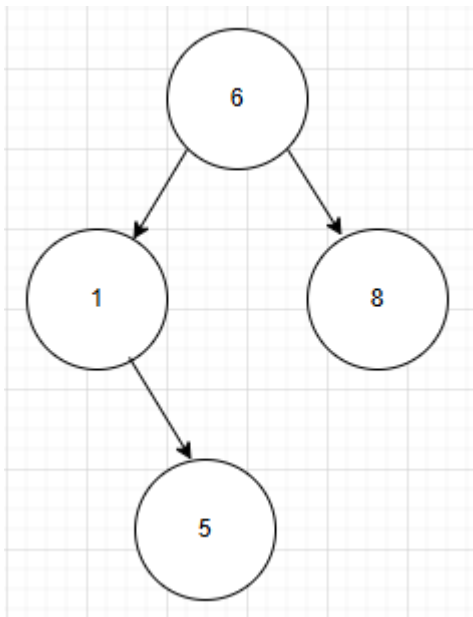
Rotación derecha en 5:



$FB(1)=0$, $FB(5)=0$, $FB(3)=0$, $FB(8)=0$, $FB(6)=+1$
Balanceado.

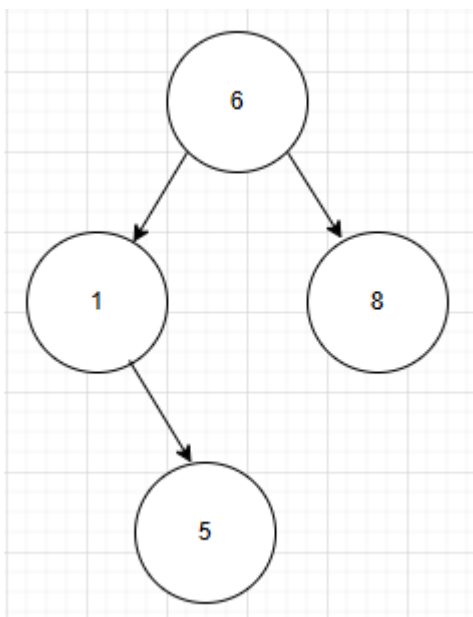
f. Eliminar 3

El nodo 3 tiene dos hijos (1 y 5) \rightarrow usamos el **mayor de los menores** \rightarrow
predecesor in-order = 1



$FB(5) = 0$, $FB(1) = -1$, $FB(6) = +1$, $FB(8) = 0$
Todos los $FB \in \{-1, 0, +1\} \rightarrow$ el árbol está balanceado.

Resultado final:



Rotaciones aplicadas:

- (c) Caso LR en 8 \rightarrow (izquierda en 5, derecha en 8)
- (e) Caso LL en 5 \rightarrow (derecha en 5)

19. Partiendo de un Árbol B vacío de orden 4, se solicita realizar las siguientes operaciones:

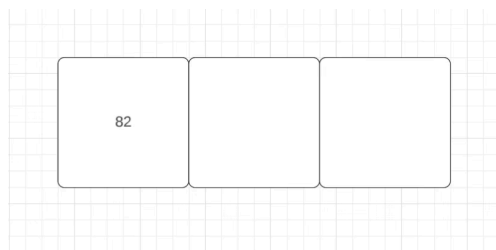
- a. Agregar el 82.**
- b. Agregar el 12.**
- c. Agregar el 102.**
- d. Agregar el 36.**
- e. Agregar el 61.**
- f. Eliminar el 82.**
- g. Eliminar el 36.**
- h. Eliminar el 102.**

RESOLUCIÓN

Secuencia: +82, +12, +102, +36, +61, -82, -36, -102

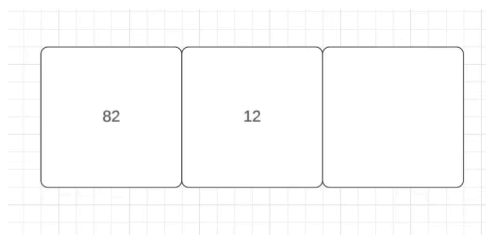
A) Insertar 82

El árbol está vacío, por lo tanto se crea un nodo raíz con el elemento 82.



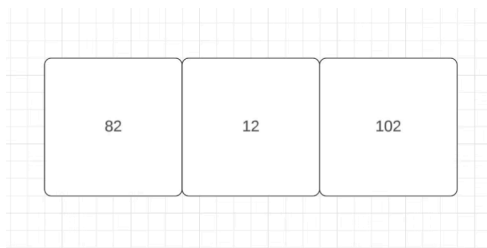
B) Insertar 12

12 es menor que 82, por lo tanto se inserta en el mismo nodo (no hay overflow, el orden es 4 y admite hasta 3 claves).



C) Insertar 102

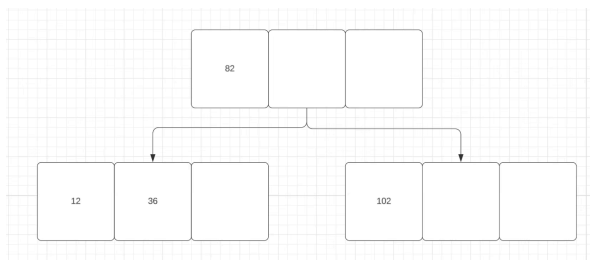
102 es mayor que 82, se agrega al mismo nodo. Aún no se excede el límite de claves (3 por nodo).



D) Insertar 36

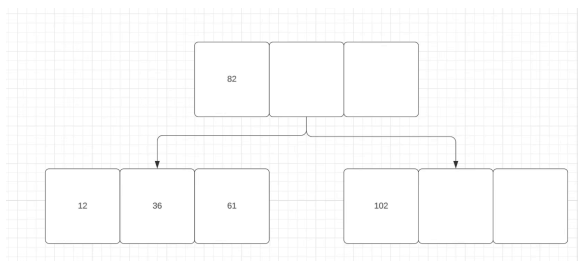
Al insertar 36, el nodo queda con 4 claves (12, 36, 82, 102), lo cual genera un overflow.

Se divide el nodo, promoviendo la clave central (82) al nuevo nodo raíz.



E) Insertar 61

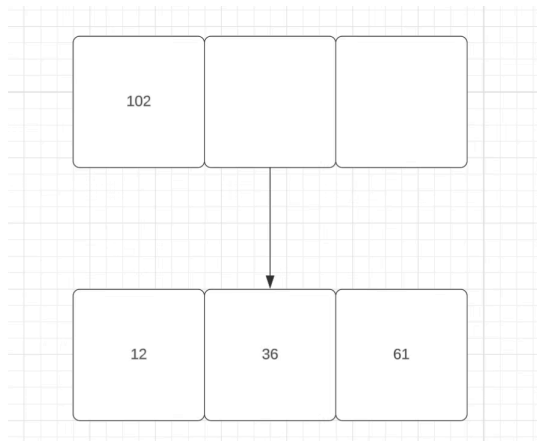
61 es menor que 82, por lo que cae en el subárbol izquierdo. Se inserta allí (12, 36, 61), sin necesidad de dividir, ya que el nodo admite hasta 3 claves.



F) Eliminar 82

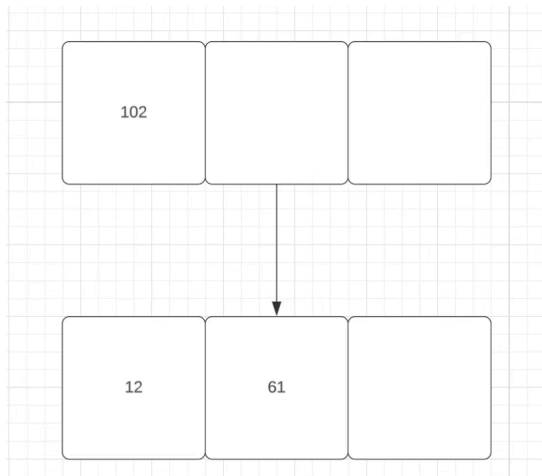
Para eliminar la clave 82 (que está en la raíz), se busca el sucesor inmediato (mínimo del subárbol derecho), que es 102.

Se reemplaza 82 por 102 y se elimina 102 del subárbol derecho.



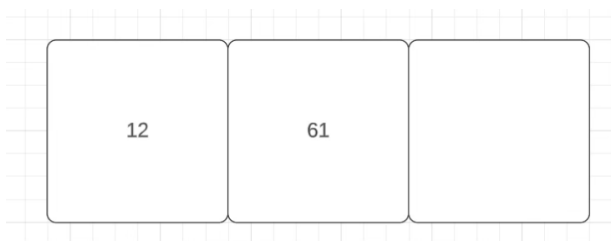
G) Eliminar 36

36 se encuentra en el nodo izquierdo. Se elimina directamente, quedando (12, 61). El nodo sigue siendo válido (posee al menos una clave).



H) Eliminar 102

Al eliminar 102, que es la raíz, se reemplaza por su hijo izquierdo. El nuevo árbol queda con un solo nodo (12, 61).



20. Partiendo de un Árbol B vacío de orden 5, se solicita realizar las siguientes operaciones:

- a. Agregar el 53
- b. Agregar el 62
- c. Agregar el 31
- d. Agregar el 105
- e. Agregar el 85
- f. Agregar el 55
- g. Eliminar el 105
- h. Eliminar el 62

Orden: 5

Máx claves por nodo: 4 (Orden - 1)

Mín claves por nodo: 2 (Máx claves / 2)

a) Inserto 53 a un Árbol B vacío

| | | | |
|----|--|--|--|
| 53 | | | |
|----|--|--|--|

b) Inserto 62

| | | | |
|----|----|--|--|
| 53 | 62 | | |
|----|----|--|--|

c) Inserto 31

| | | | |
|----|----|----|--|
| 31 | 53 | 62 | |
|----|----|----|--|

d) Inserto 105

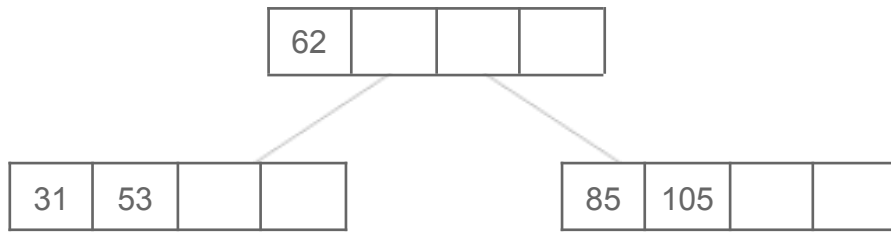
| | | | |
|----|----|----|-----|
| 31 | 53 | 62 | 105 |
|----|----|----|-----|

e) Inserto 85

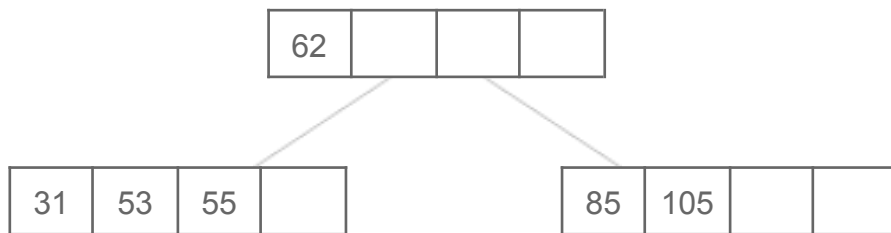
1) Ordeno con un elemento *dummy* para realizar un *Split*

| | | | | |
|----|----|----|----|-----|
| 31 | 53 | 62 | 85 | 105 |
|----|----|----|----|-----|

2) Realizo el *Split*

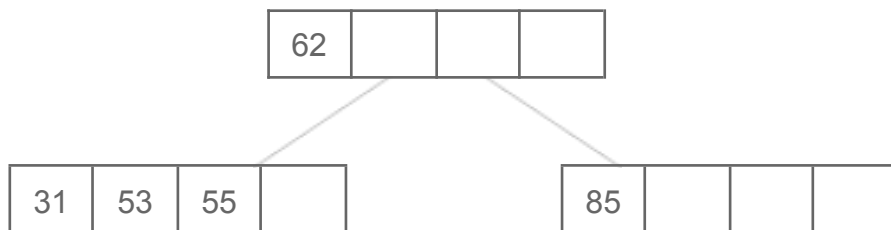


f) *Inserto el 55*

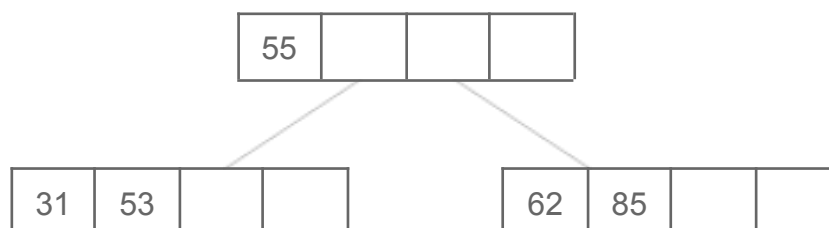


g) *Elimino el 105*

- 1) El hijo derecho queda con una sola clave, se debe realizar una *donación* desde el hermano izquierdo para cumplir con el mínimo requerido de claves

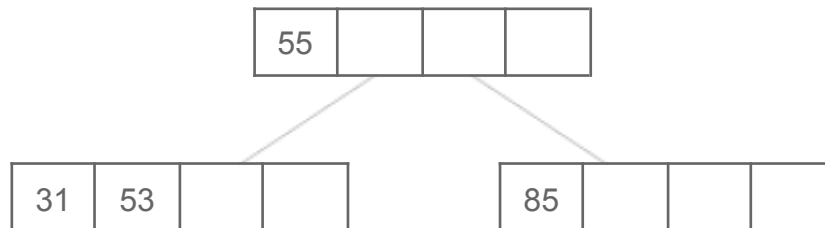


- 2) El padre da el 62 al hijo derecho, el hermano izquierdo da el 55 al padre



h) *Elimino el 62*

- 1) El hijo derecho queda con una sola clave, por lo que se debe realizar una fusión porque el hermano izquierdo no puede donar (quedaría por debajo de la cantidad mínima de claves)



- 2) Finalmente, nos quedamos solo con el nodo raíz



Resultado Final



Se realizaron inserciones con *Split*, y en las eliminaciones se aplicaron primero *donación* y luego *fusión* para mantener las propiedades del Árbol B.