## -Binario de búsqueda

**Insertar:**

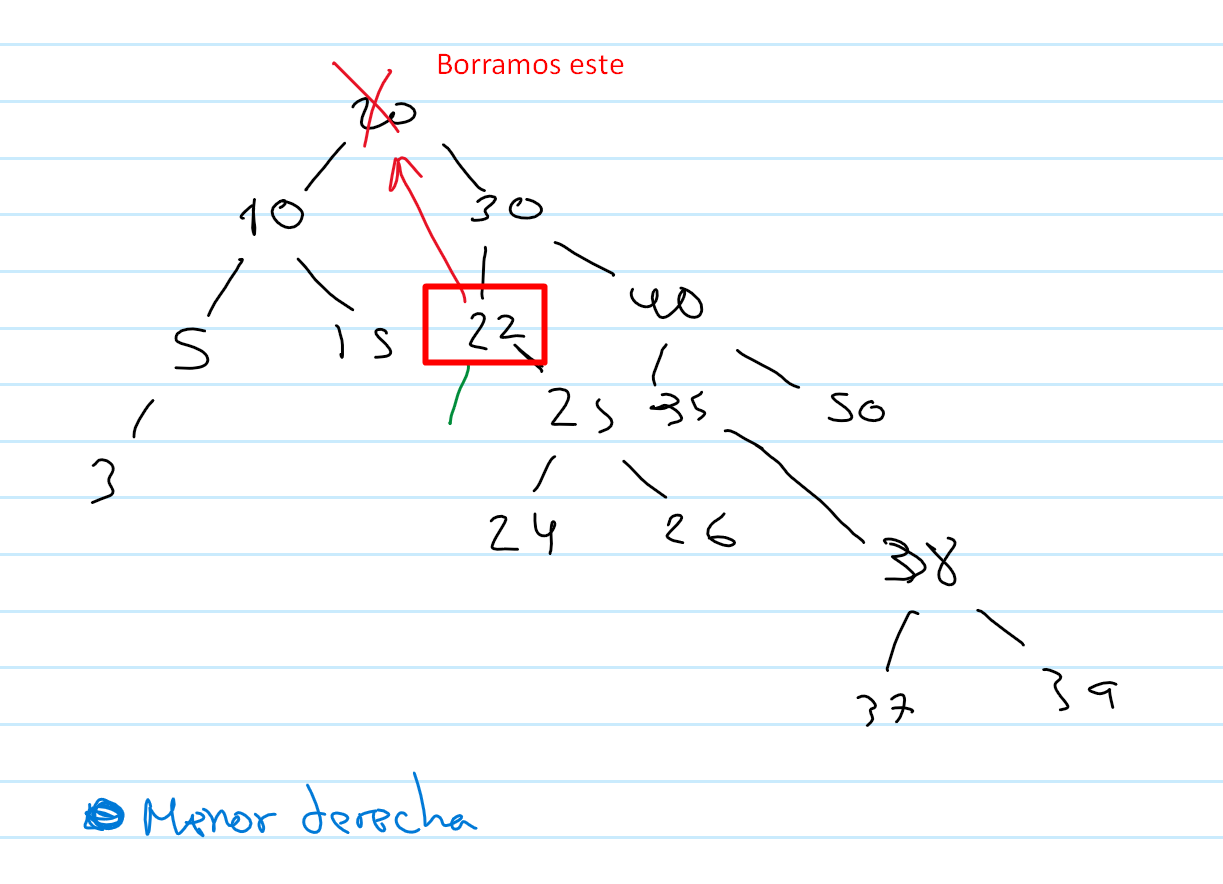
En los árboles binarios la inserción simplemente se hace mirando si es menor va a la izquierda, si es mayor va a la derecha y ya. (sin ningún tipo de ordenado ni rotación ni nada, como si queda super largo por su izquierda).

**Borrar:**

El nodo que borramos se elige el método mayor de la izquierda o menor de la derecha.

Si es menos de la derecha se coje literalmente al menor de su subarbol derecho y me convierto en el. Si ese menor tenia 1 solo hijo ese hijo ocupa la posicion de ese menor, si tiene dos hijos lo reemplaza su hijo mayor.

(en arboles binario no hay rotaciones ni nada, solo la cosa de la sustitucion en el borrado)

****

Si en este ejemplo hemos elegido el 22, entonces su hijo (el 25) pasará a ocupar su posición.

Aquí solo se intercambio por ese, y ya, no hay que hacer ni rotacion ni combinaciones ni nada, se deja asi.

## -AVL

**Preorden: raiz , izquierda ,derecha**

**Postorden: Izquierda, derecha, y raiz**

**Inorden: Izquierda , la raiz , y derecha**

**Insertar:**

Primero se inserta, y luego se calcula su factor de equilibrio (altura subarbol derecha menos altura subarbol izquierda, si dá 2 o -2 entonces hay que reestructurar con DD, DI, II , ID.

**Borrar:**

Primero se borra el elemento, luego si tiene hijos se sustituye por su menor de la derecha o por su mayor de la izquierda.

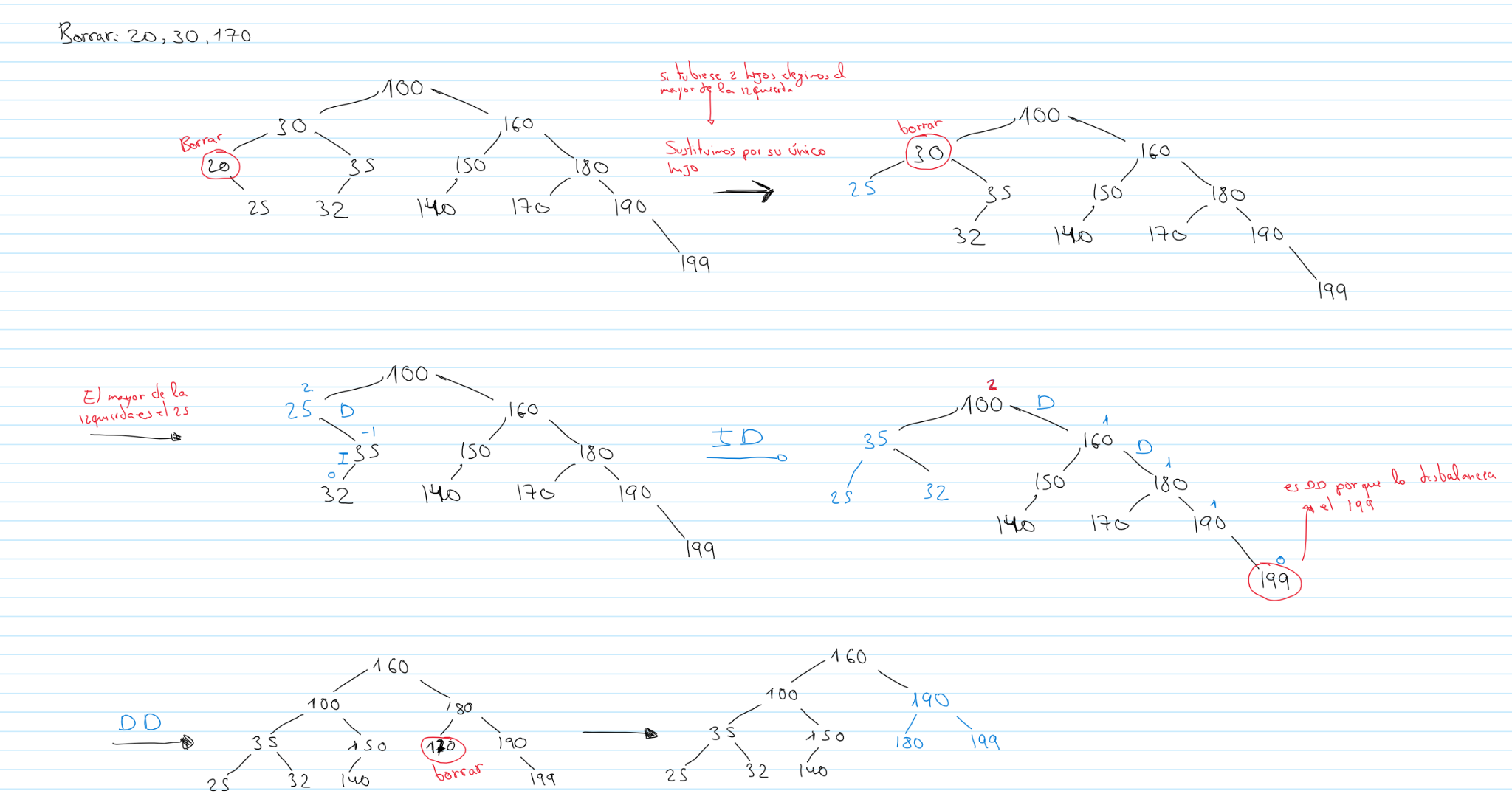
-Si tiene 1 hijo: ese hijo se sube

-Si tiene 2 hijos:

-Si es mayor de la izquierda, se sube ese

-Si es menor de la derecha, se sube ese

Segundo, después de haber sustituido por su hijo (según algoritmo si izq o der) rotamos como si acabasemos de insertar: DD, DI , II , o ID

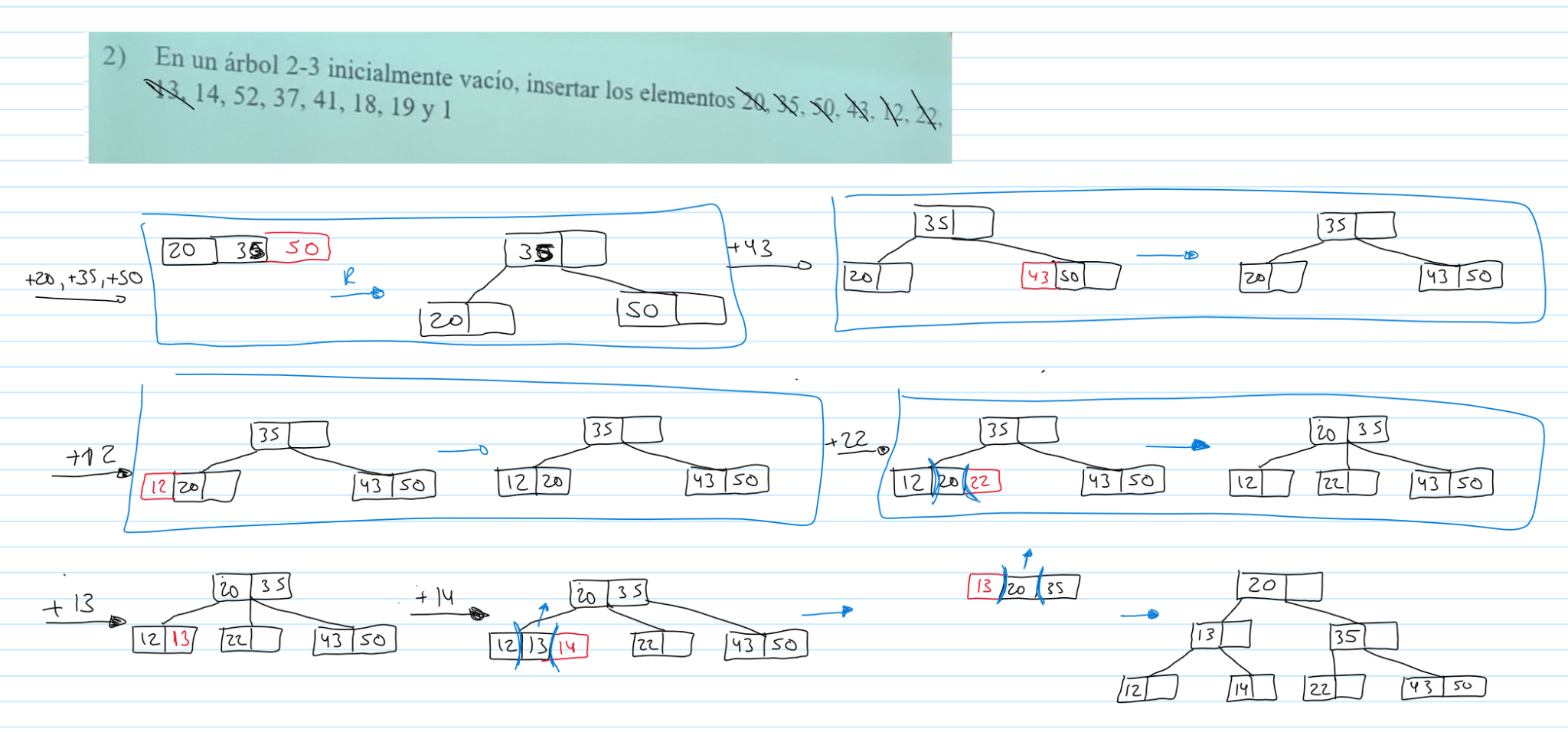


La rotacion del 25 35 y 32 en la imagen no es correcta

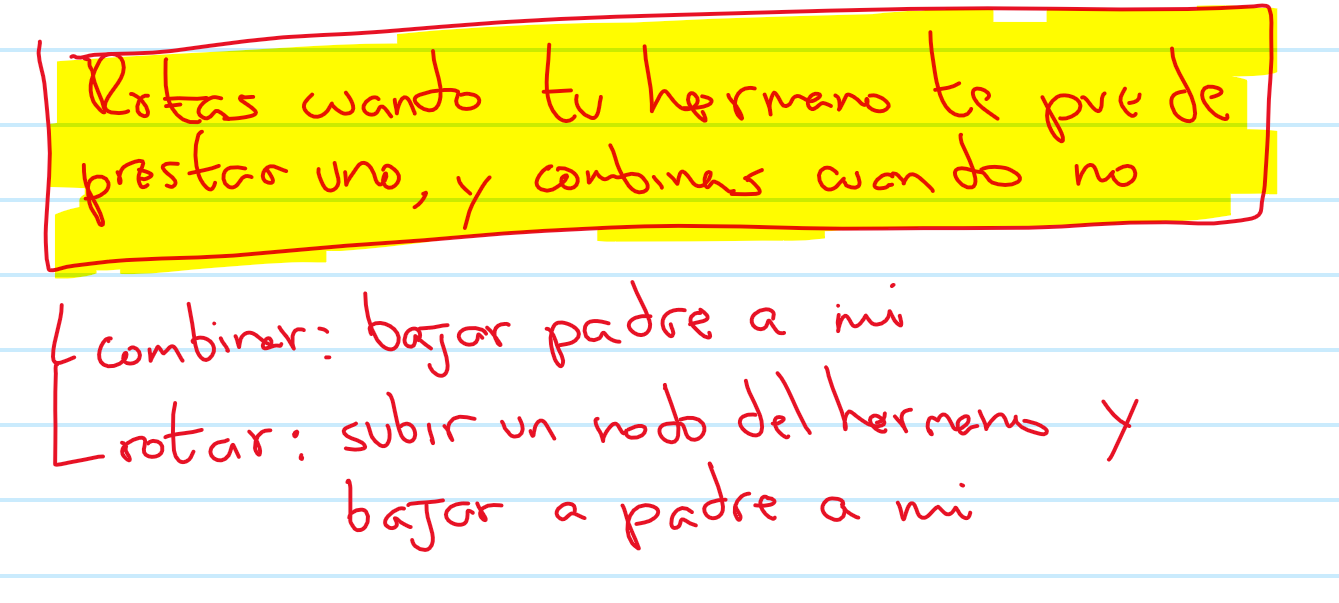
## -2-3

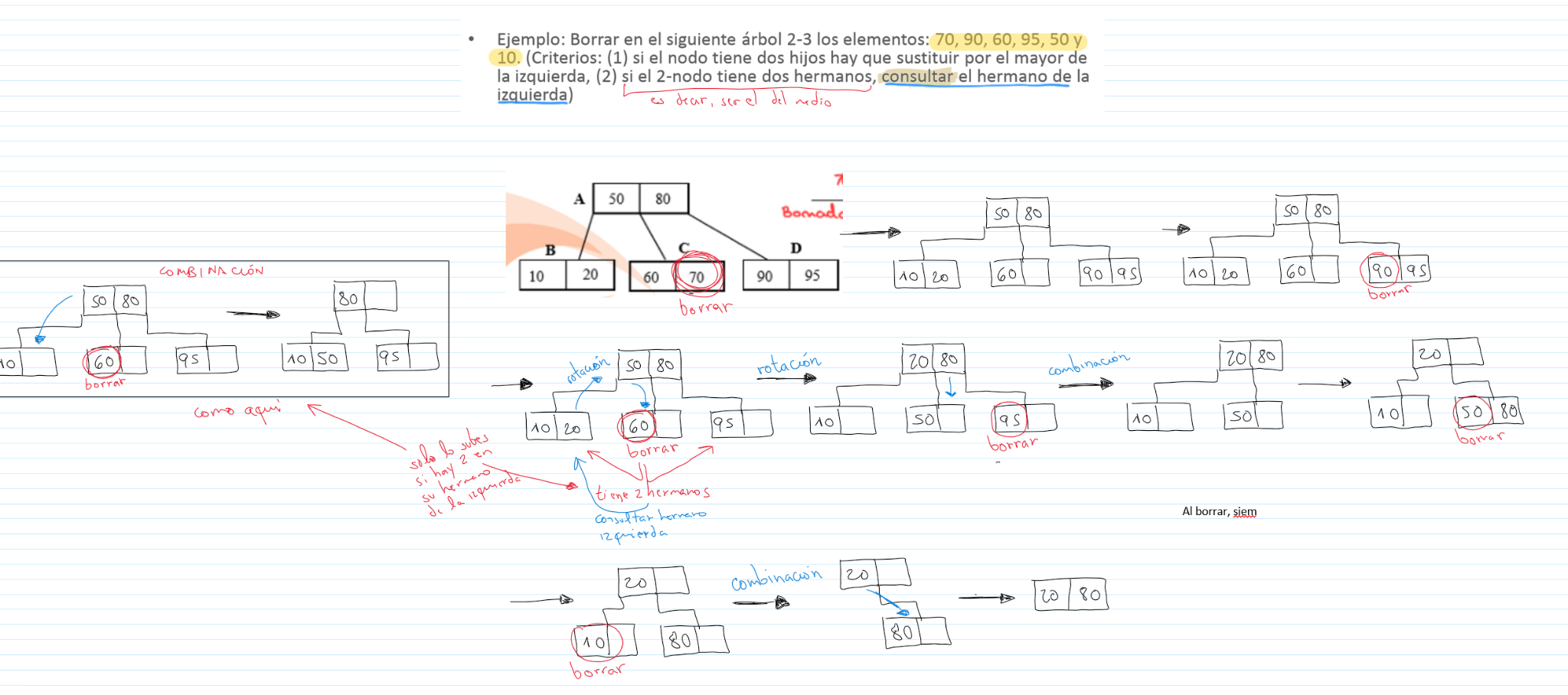
**Insertar:**

**-**Solo se inserta en las hojas (al principio la raíz es hoja)



**Borrar:**

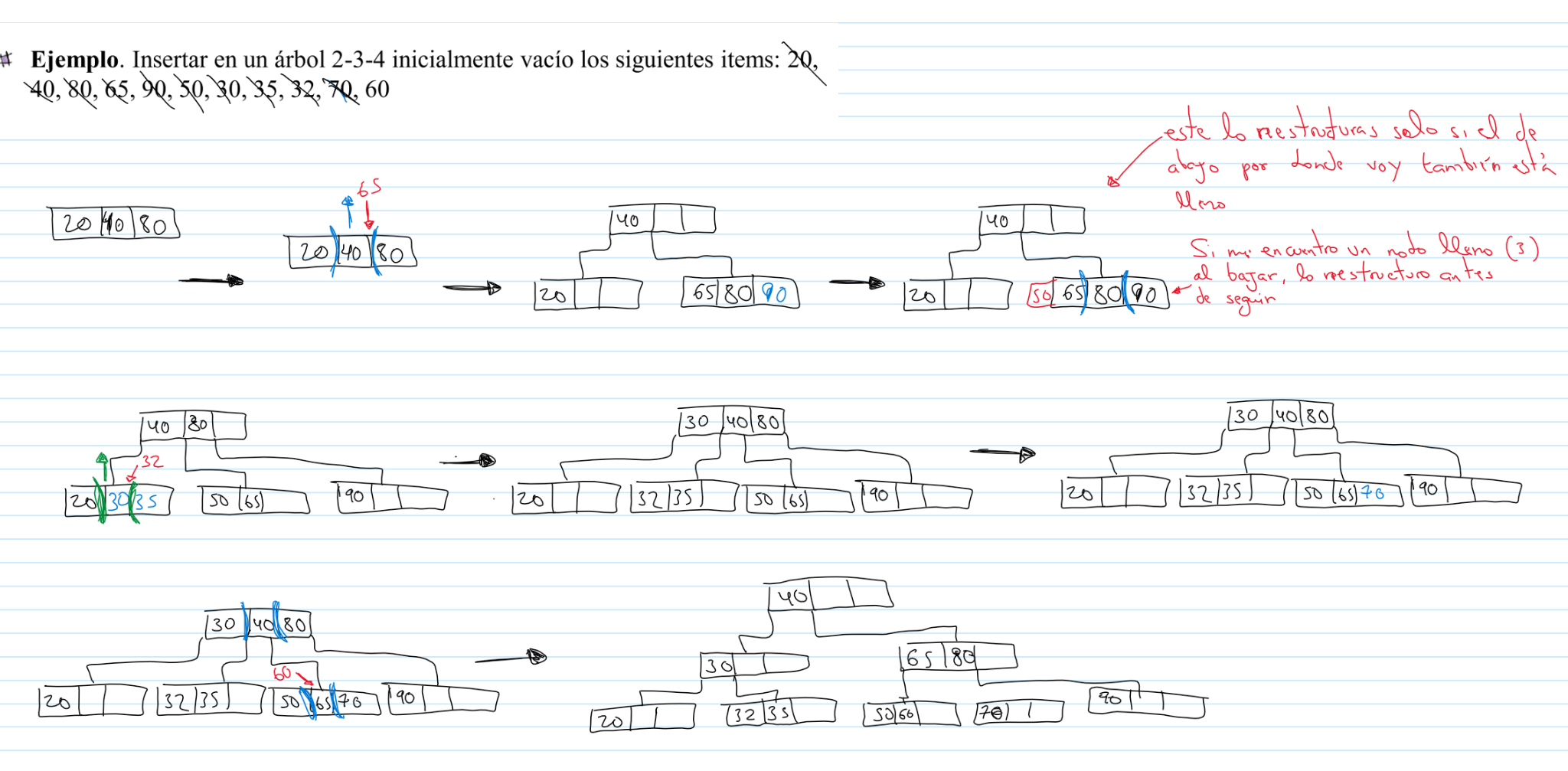
****

****

## -2-3-4

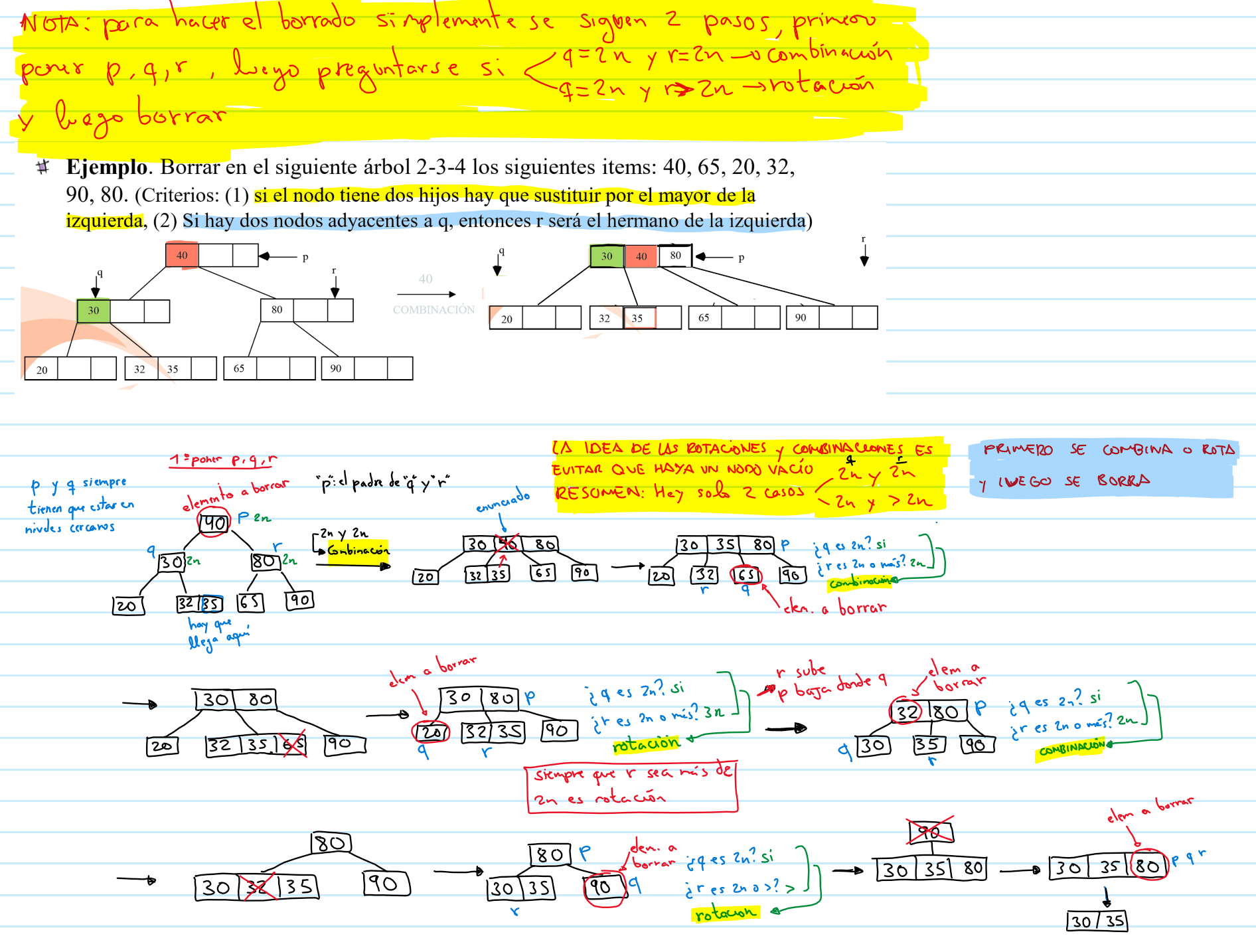
**Insertar:**

Lo mismo que con el 2-3 pero con 1 posición más. Se inserta en la raiz siempre.

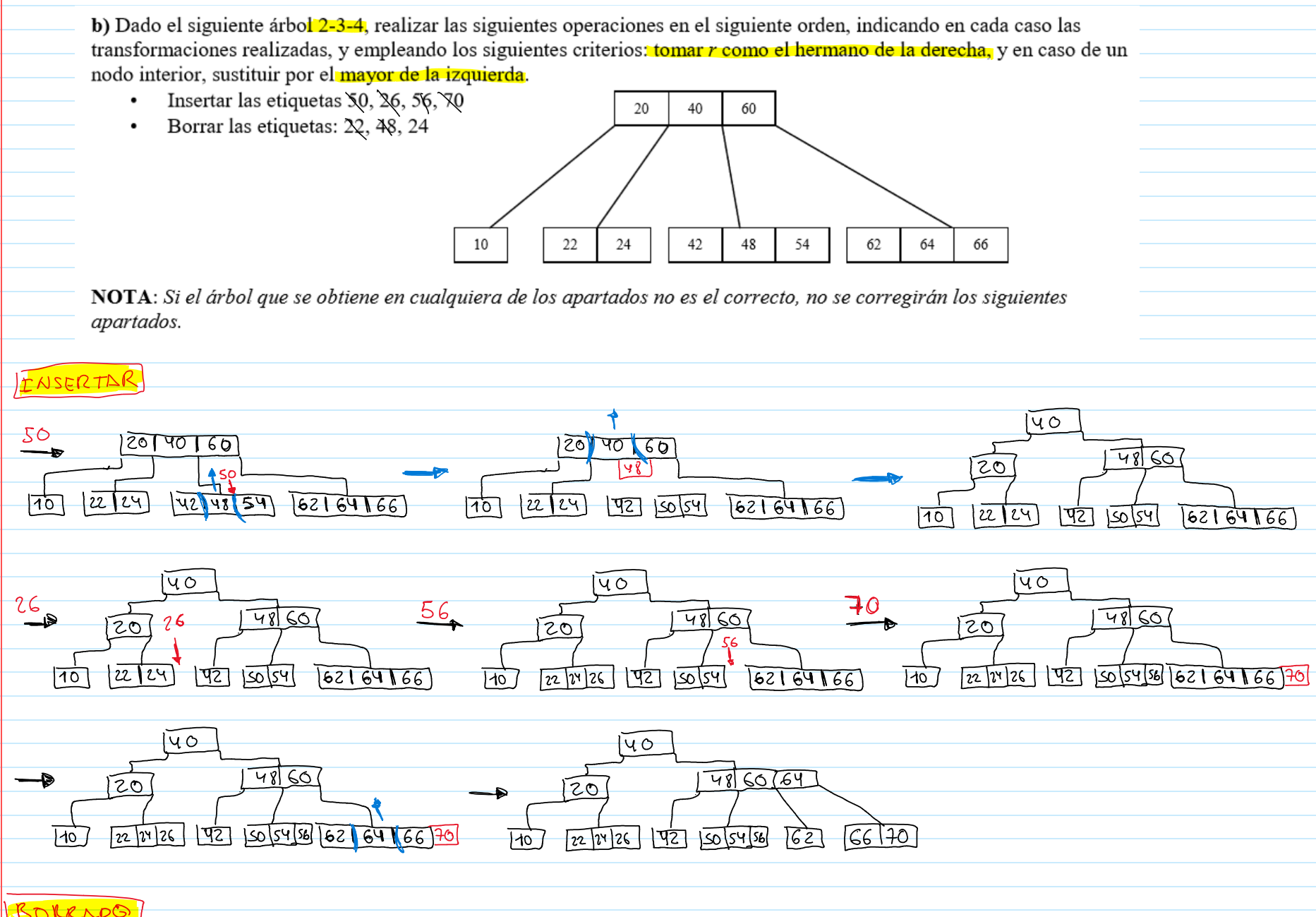
****

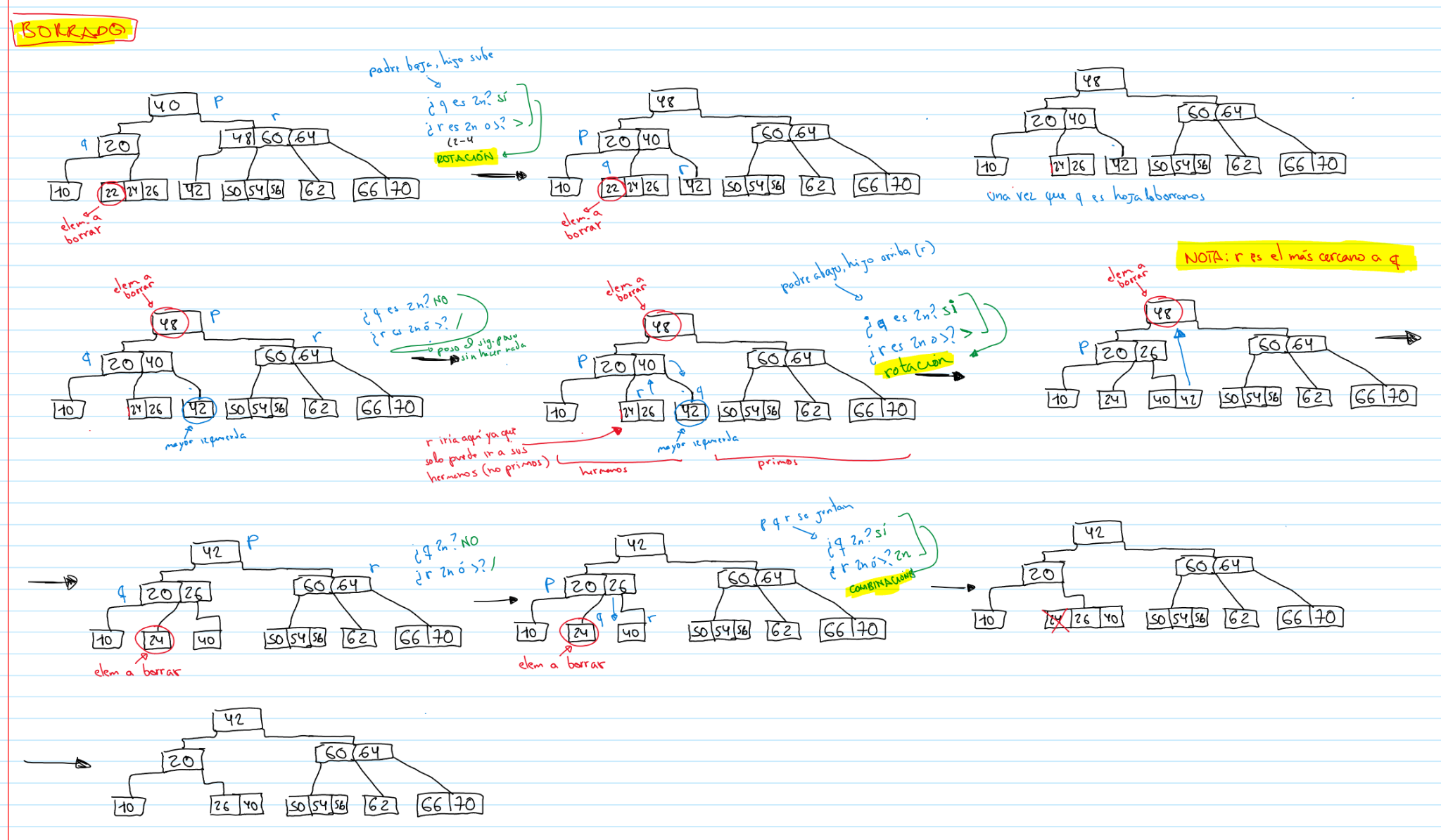
**Al pasar por un 4 nodo al insertar un número, se reestructura primero este 4 nodo**

**Borrar:**

****

**Ejercicio util 234**

****

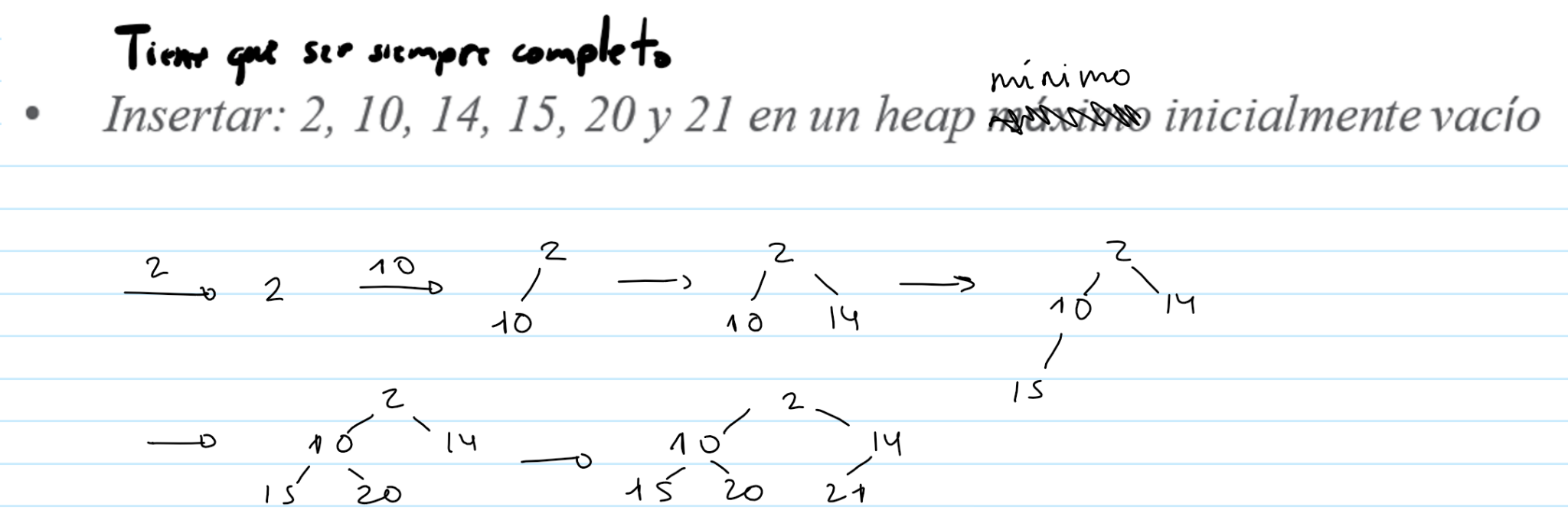
****

## -HEAP

Este árbol tiene que estar siempre completo

-MÍNIMO

Insertar:



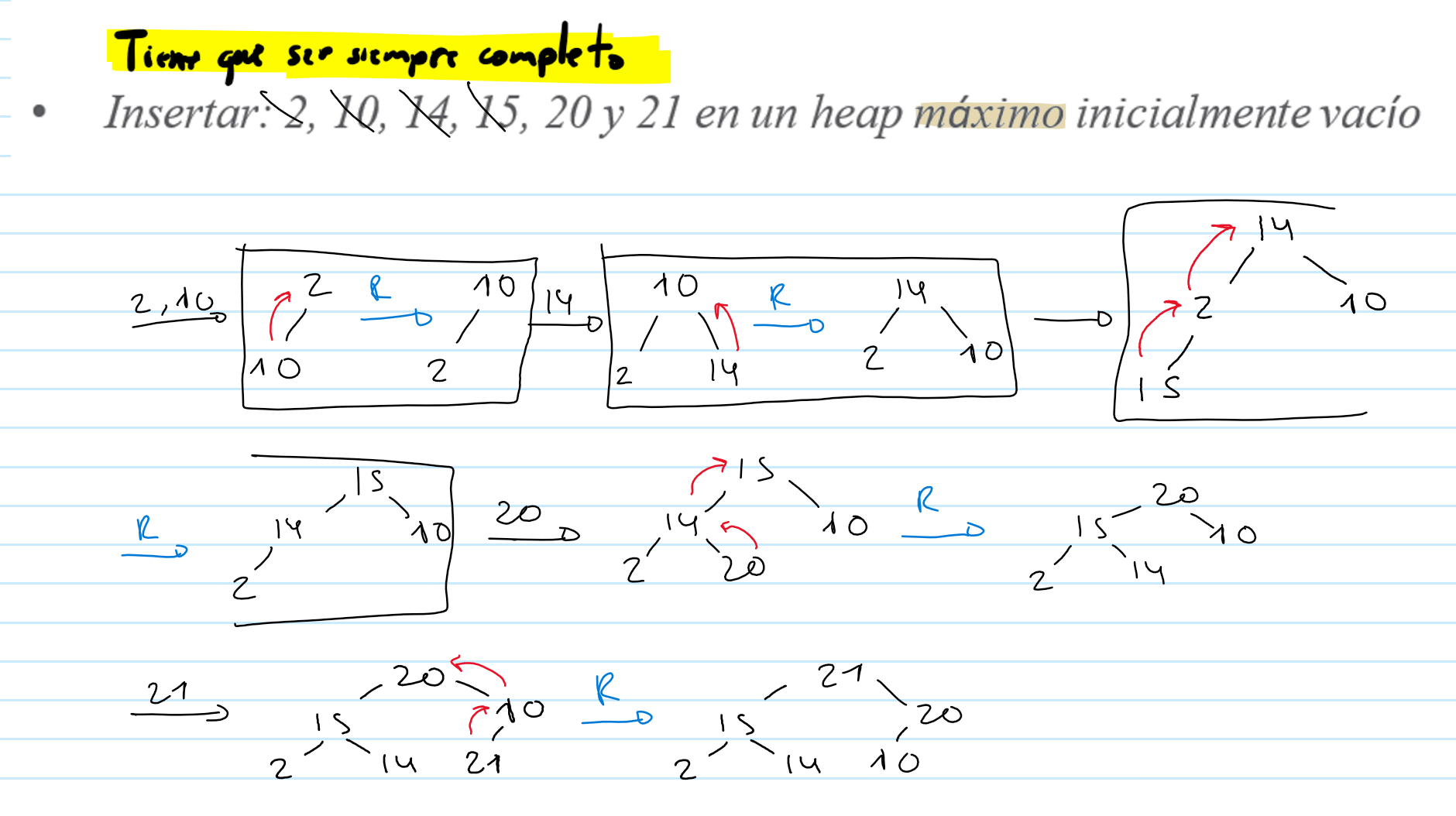
-borrar:

**EL BORRADO SOLO SE HACE SOBRE LA RAÍZ**

**Al borrar la raiz , la sustituimos por el nodo hoja más bajo de la derecha**

-MÁXIMO

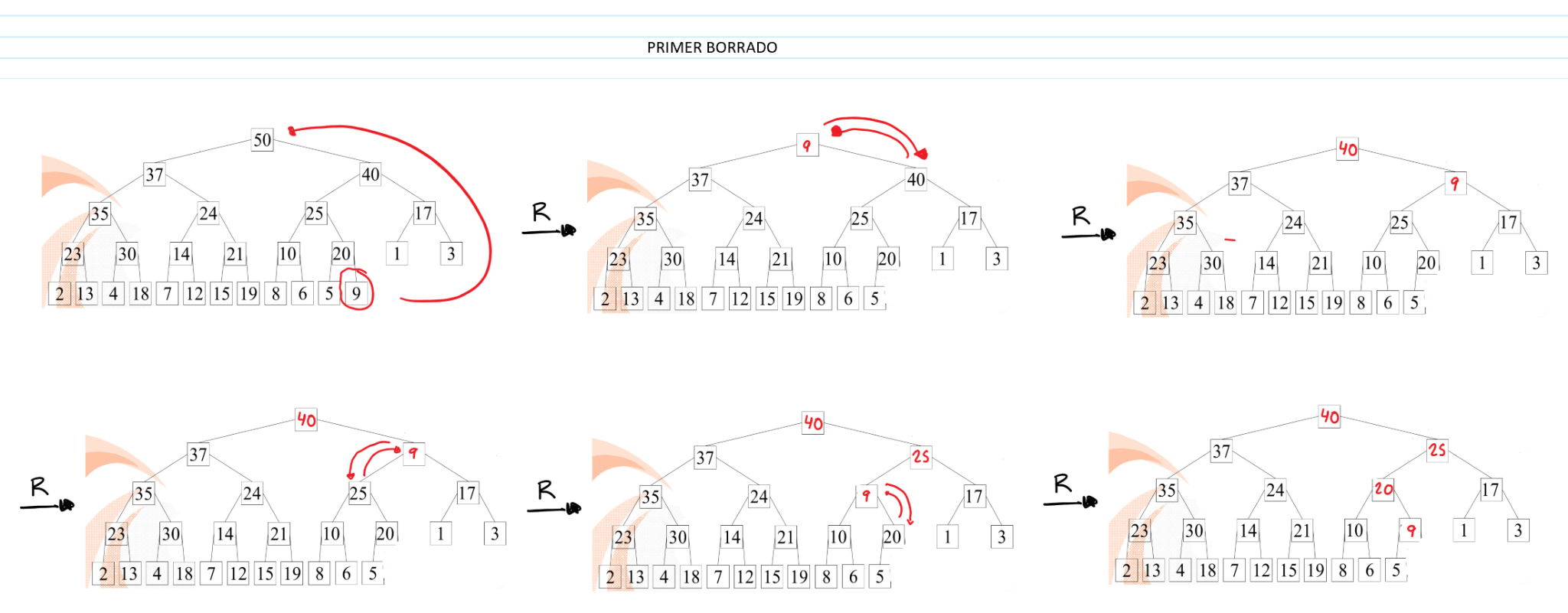
-insertar:



-borrar:

**EL BORRADO SOLO SE HACE SOBRE LA RAÍZ**

Aquí es borrar la raiz, coger el más bajo primero (que puede ser más bajo el subarbol de la izquierda o la derecha, vamos al mas bajo) ) y luego en ese subarbol el más a la derecha y sustituirlo y luego reordenar

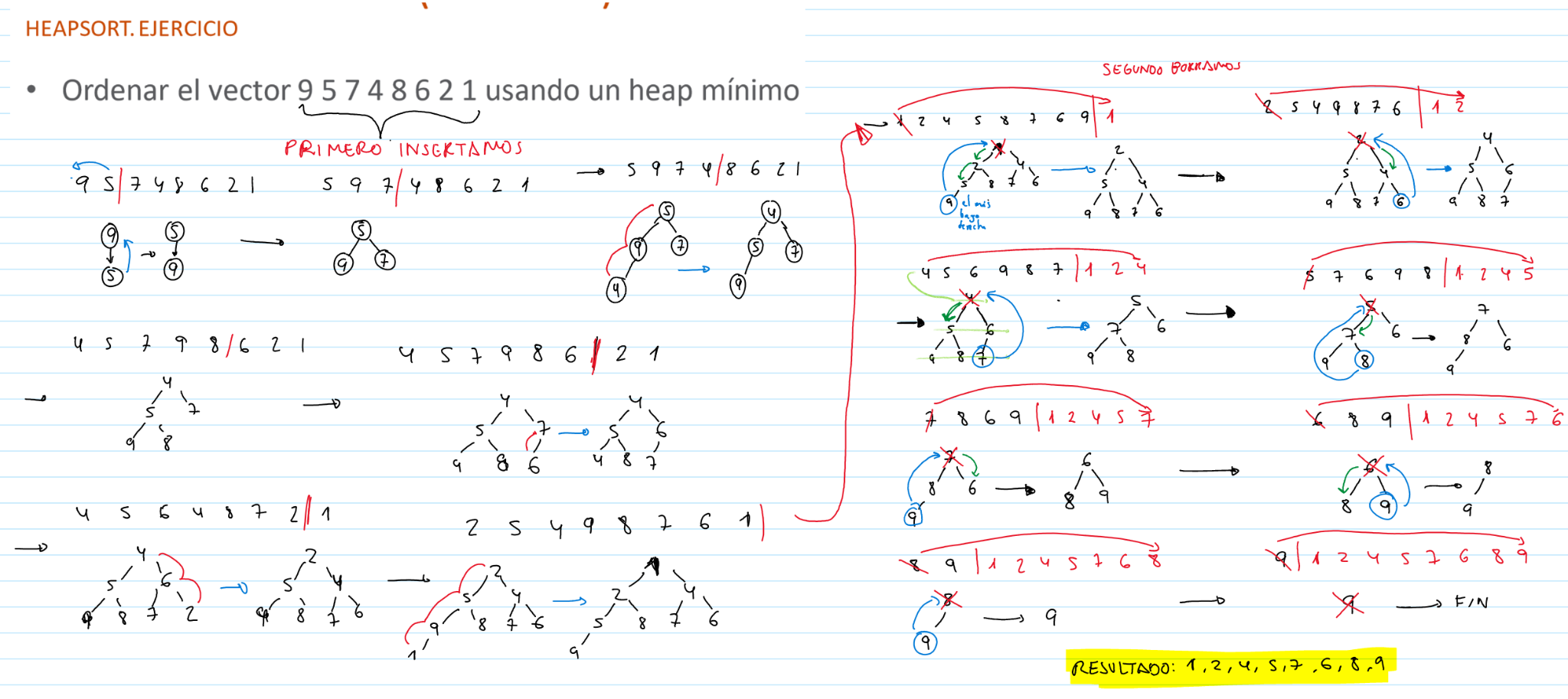
****

****

## HEAPSORT

Primero se hace la insercion, y luego el borrado

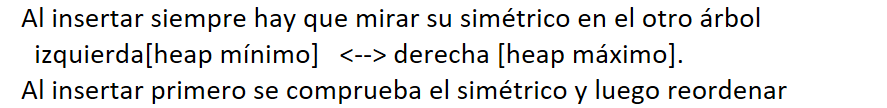
Cada vez que insertamos o borramos reordenamos lo que hay antes de la barra por niveles desde arriba abajo

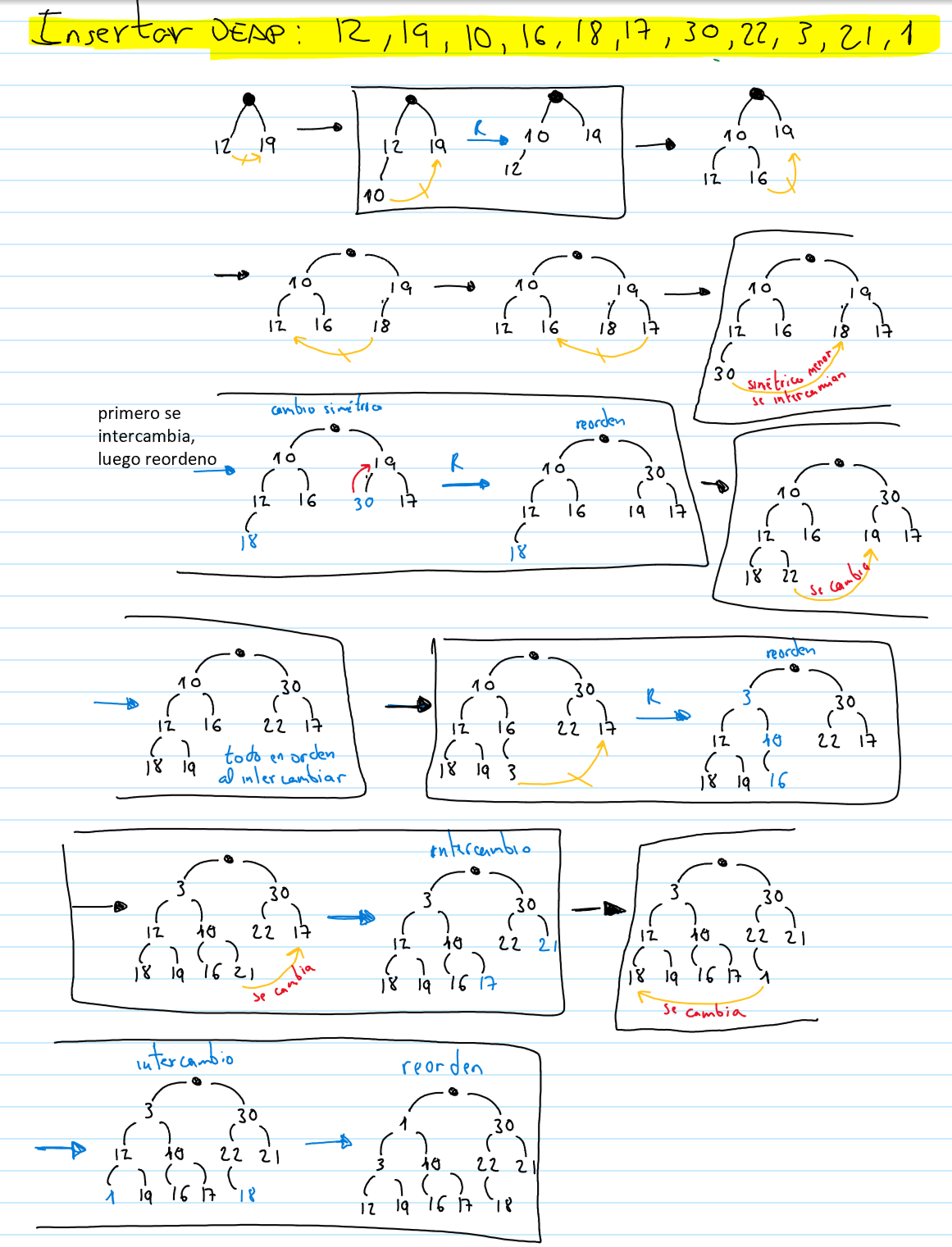


## 

## -DEAP

Insertar





Borrar

Se borra siempre la raíz propia, sea del heap minimo (izquierda) o del heap máximo (derecha)

