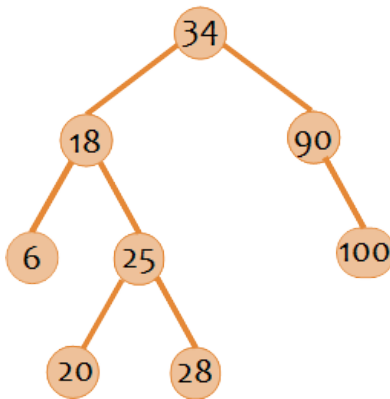
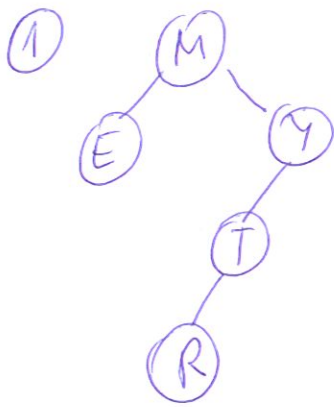


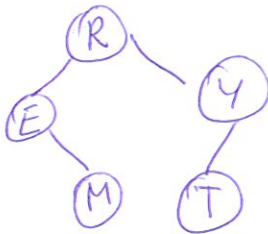
1. Para cada una de las siguientes listas de letras:
  - M, Y, T, E, R
  - R, E, M, Y, T
  - T, Y, M, E, R
  - C, O, R, N, F, L, A, K, E, S
  - a. Dibuja el árbol binario de búsqueda que se construye cuando las letras se insertan en el orden dado.
  - b. Realiza recorridos inorden, preorden y postorden del árbol y muestra la secuencia de letras que resultan en cada caso.
2. Construye un árbol binario de búsqueda para almacenar los datos 12, 8, 7, 16 y 11.
3. Explica gráficamente cómo eliminar los nodos 34, 90 y 6 del siguiente árbol binario de búsqueda:



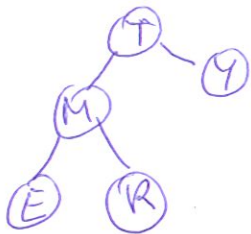
4. Dado un array de N valores, se puede construir un montículo binario de máximos (*max-heap*) “hundiendo” cada nodo interno hacia su posición.
  - a. A partir del vector [10, 2, 7, 6, 5, 9, 12, 35, 22, 15, 1, 3, 4], crea el árbol binario representado por él, siguiendo la estructura de montículo.
  - b. Este árbol binario es completo, está relleno hacia la izquierda, pero sus nodos no cumplen la relación de orden parcial. Realiza el proceso de **heapify** analizando los nodos internos y “hundiénolos” hacia su posición correcta, comenzando por el nivel n-1 (n=altura del árbol). Ten en cuenta que es un montículo de máximos (*max-heap*), por lo que en la raíz debe resultar el valor máximo.
  - c. Dibuja el montículo después de insertar los valores 16, 20 y 45.
  - d. Dibuja el montículo después de 3 operaciones sucesivas de eliminación.
5. El algoritmo *heapsort* ordena los elementos de un vector después del proceso anterior, pues podemos ver que no está totalmente ordenado. La operación que se realiza es ir eliminando el elemento máximo (raíz) tal como está en los apuntes e ir almacenándolo en un vector en el que al final estarán ordenados los elementos de mayor a menor si es un *max-heap* y de menor a mayor si es un *min-heap*.  
**Repite el ejercicio 4 sustituyendo el *max-heap* con un *min-heap* y a continuación aplica el algoritmo *heapsort* para demostrar que los elementos quedan ordenados de menor a mayor.**



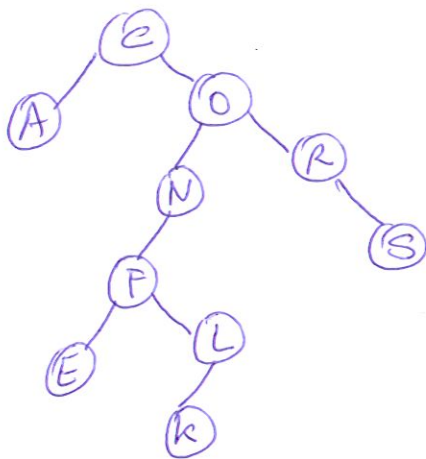
Inorden (IRA): E-M-R-T-Y  
 Preorden (RID): M-E-Y-T-R  
 Postorden (IDR): E-R-T-Y-M



Inorden: E-M-R-T-Y  
 Preorden: R-E-M-Y-T  
 Postorden: M-E-T-Y-R



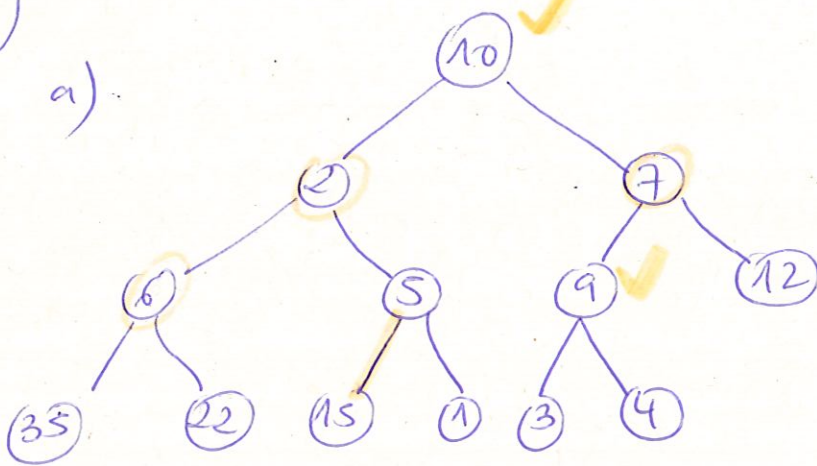
Inorden: E-M-R-T-Y  
 Preorden: T-M-E-R-Y  
 Postorden: E-R-M-Y-T



Inorden: A-C-E-F-K-L-N-O-R-S  
 Preorden: C-A-O-N-F-E-L-K-R-S  
 Postorden: A-E-K-L-F-N-S-R-O-C

4)

a)

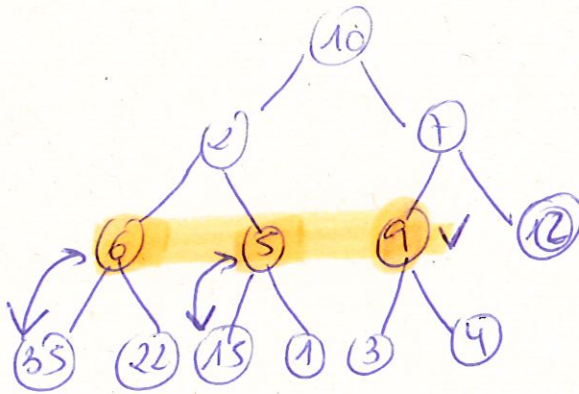


max-heap  $\rightarrow$  orden parcial

min-heap  $\{ \text{clave} \leq \text{hijos} \}$

max-heap  $\{ \text{clave} \geq \text{hijos} \}$

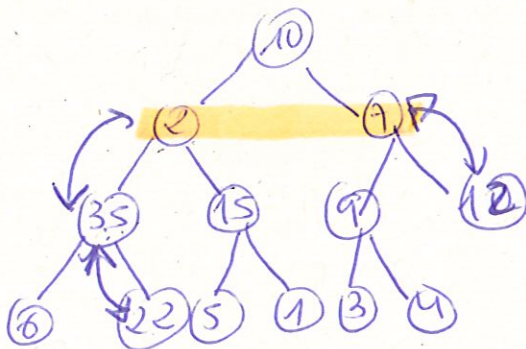
b) NO CUMPLEN ORDEN PARCIAL max-heap!  $\{ \text{clave} \geq \text{hijos} \}$   
 HEAPIFY {  
 Se comienza con el último nodo INTERNO  
 Se intercambia con su hijo mayor  
 Se repite el nodo intercambiado



-  $6 > 3$  y  $9 > 4 \Rightarrow$  no hay cambio

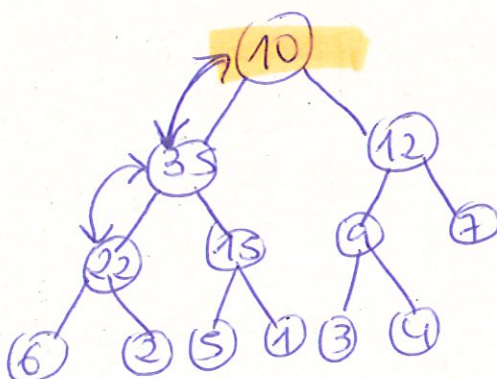
-  $\max(5, 15, 1) = 15 \Rightarrow$  intercambio

-  $\max(6, 35, 22) = 35 \Rightarrow$  intercambio

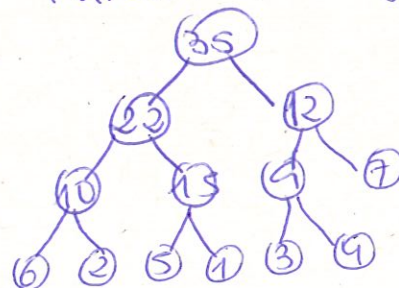


-  $\max(2, 9, 12) = 12 \Rightarrow$  intercambio y propaga

-  $\max(2, 35, 15) = 35 \Rightarrow$  intercambio y propaga



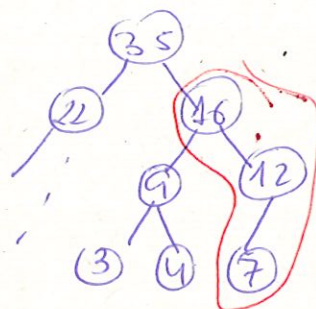
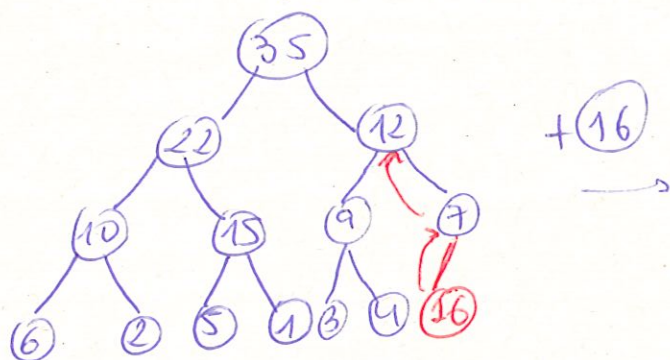
$\max(10, 35, 12) = 35 \Rightarrow$  intercambio y propaga



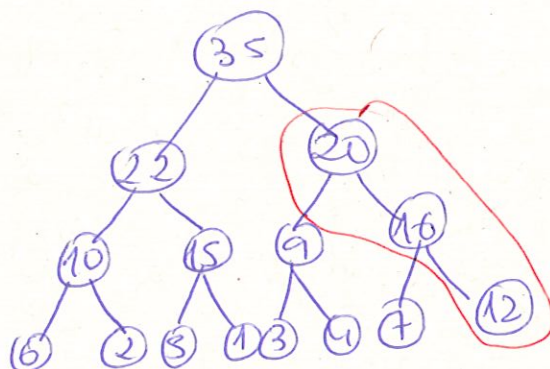
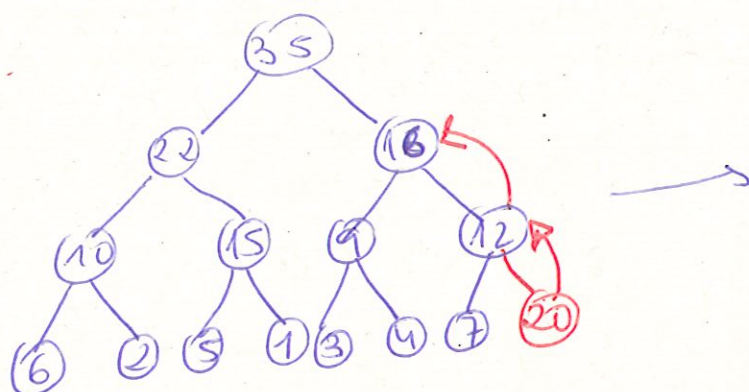
[35, 22, 12, 10, 15, 9, 7, 6, 2, 5, 1, 3, 4]



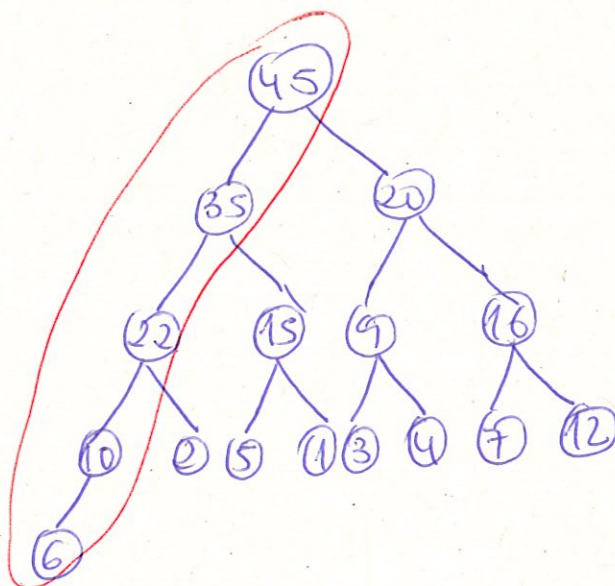
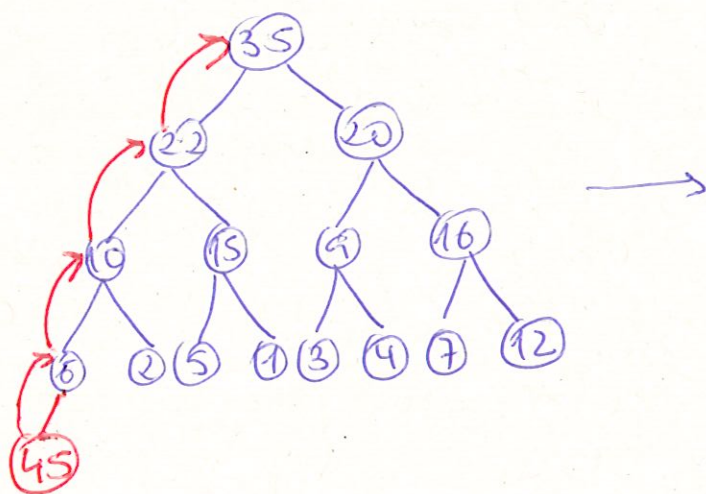
⑥ Insertar ~~16~~ → como hoja ~~178~~ y se hace "floatar"



Insertar 20

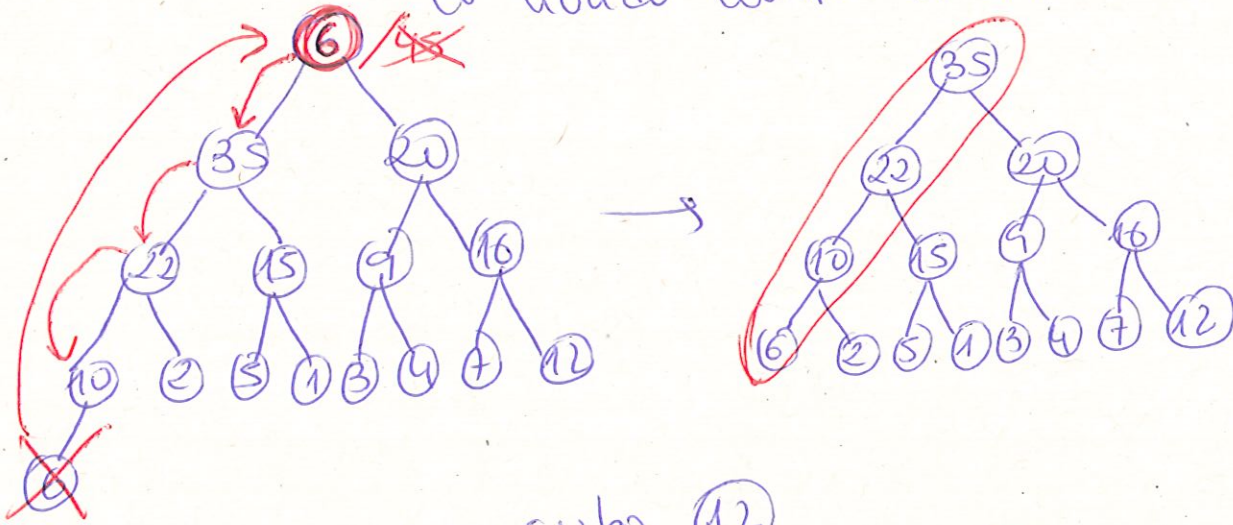


Insertar 45

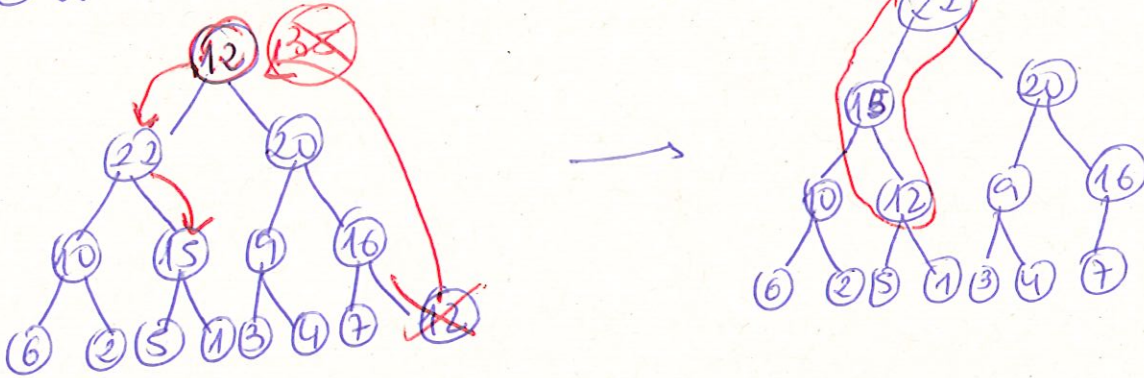


# ① Tres operaciones sucesivas de eliminación

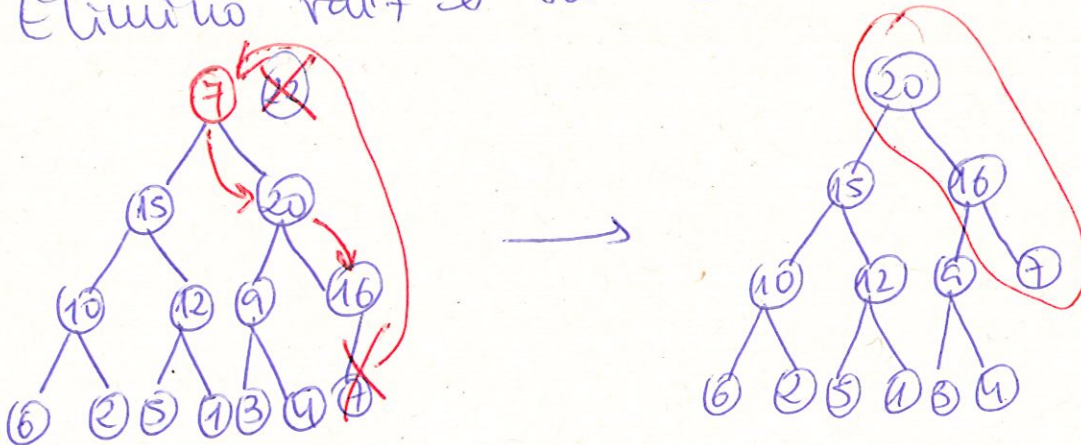
Suprimo raíz y subo hijo izquierdo  
lo fundo hasta su hijo



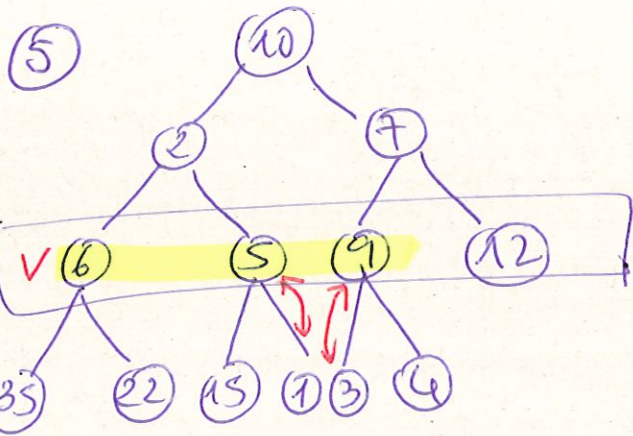
Elimino raíz y subo 12



Elimino raíz y subo 7







min-heap es orden parcial  
donde  $\leq$  hijos

HEAPIFY: se comienza con el último nodo interno,  
se intercambia con su hijo menor y se propaga

**NIVEL 3**

$\min(9, 3, 4) = 3 \Rightarrow$  intercambio

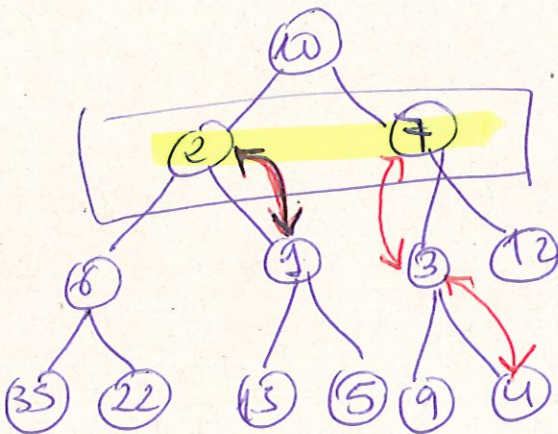
$\min(5, 15, 1) = 1 \Rightarrow$  intercambio

$\min(6, 35, 22) = 6 \Rightarrow$  NO HAY CAMBIO

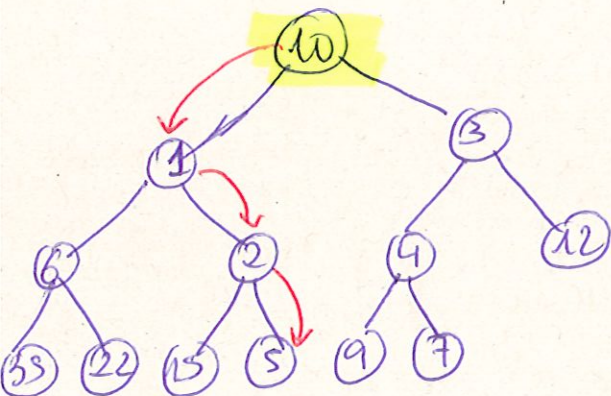
**NIVEL 2**

$\rightarrow \min(2, 6, 1) = 1 \Rightarrow$  intercambio

$\downarrow$   
 $\min(2, 5, 15) = 2$   
 $\downarrow$  no se propaga



$\rightarrow \min(7, 3, 12) = 3 \Rightarrow$  intercambio  
se propaga  
 $\hookrightarrow \min(7, 4, 9) = 4 \Rightarrow$  intercambio



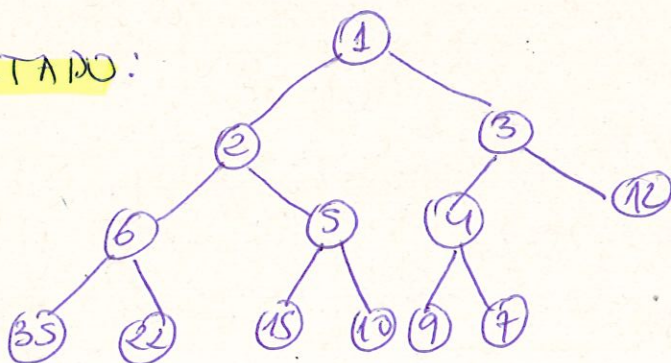
**NIVEL 1**

$\min(10, 1, 3) = 1 \Rightarrow$  intercambio

$\hookrightarrow \min(10, 6, 2) = 2 \Rightarrow$  propaga

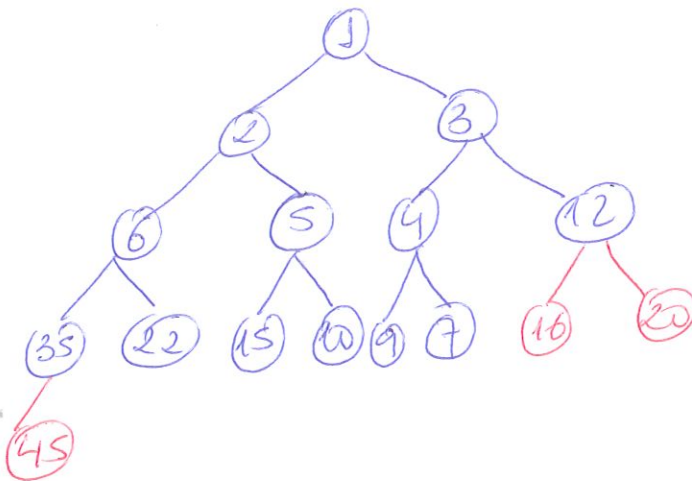
$\hookrightarrow \min(10, 15, 5) = 5 \Rightarrow$  propaga

**RESULTADO:**





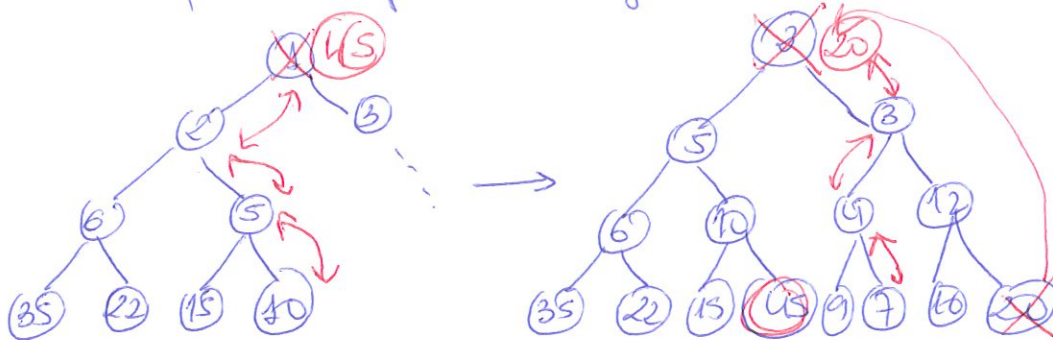
c) Dibuja el maxículo después de insertar 16, 20 y 45



NO HAY CAMBIOS,  
SE CUMPLE LA REGLA  
DE ORDEN PARCIAL

d) HEAPSORT → 3 operaciones sucesivas de eliminación

\* Se suprime la raíz (1) y se sustituye por el último elemento (45), hundiéndolo a una posición que cumpla la regla de orden parcial



Suprimo raíz (2)  
y sube (20),  
y se "hunde"

