

ED – Seminario

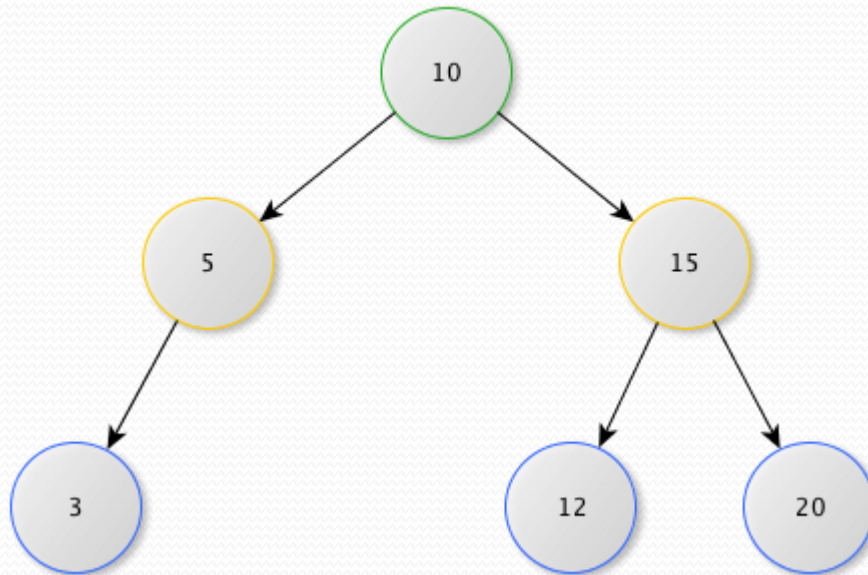
03/11/2016

Arboles de búsqueda binarios

María del Rosario Suárez Fernández

Árbol de búsqueda binario

- Cada nodo puede tener cero, uno o dos hijos como máximo
- Los valores de los nodos no se repiten



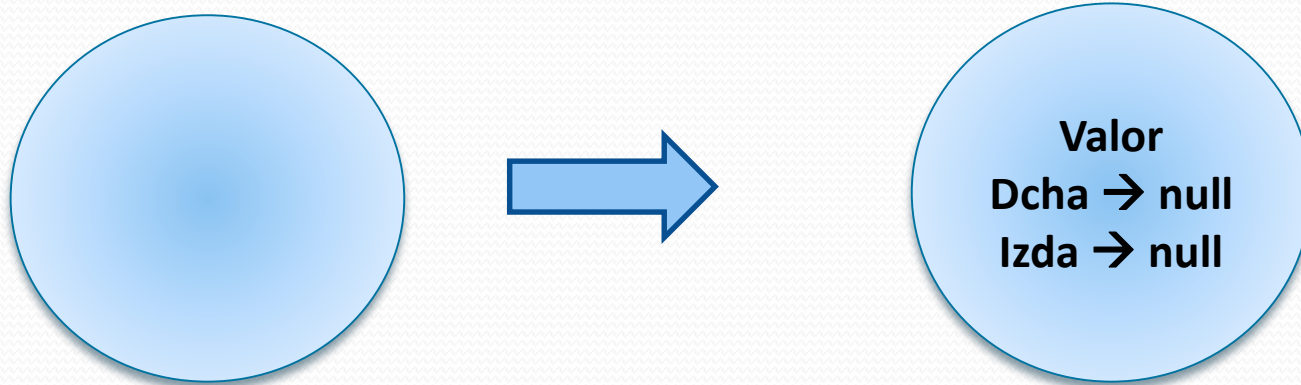
Añadir un nodo nuevo

- Para cada elemento que se inserta, si este es mayor que la raíz se inserta en el subárbol derecho y si es menor en el subárbol izquierdo
- Cuatro casos
 - Árbol vacío
 - El nodo ya existe
 - El nodo es mayor que la raíz
 - El nodo es menor que la raíz



Procedimiento
recursivo

Insertar. Árbol vacío



Hacer que la raíz sea un nuevo **BSTNode<T>**

Información → valor proporcionado
Nodo izquierdo → null
Nodo derecho → null

Insertar. Valor menor que la raíz

Insertar el nodo con valor 2

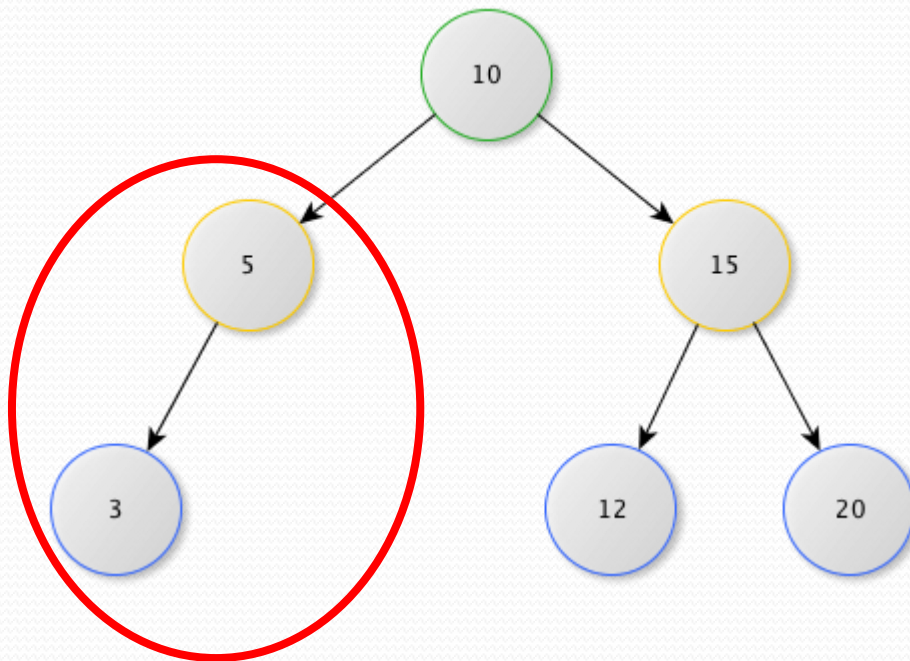


```
Raíz=10  
Valor=2  
add(Raíz=10, Valor=2)
```

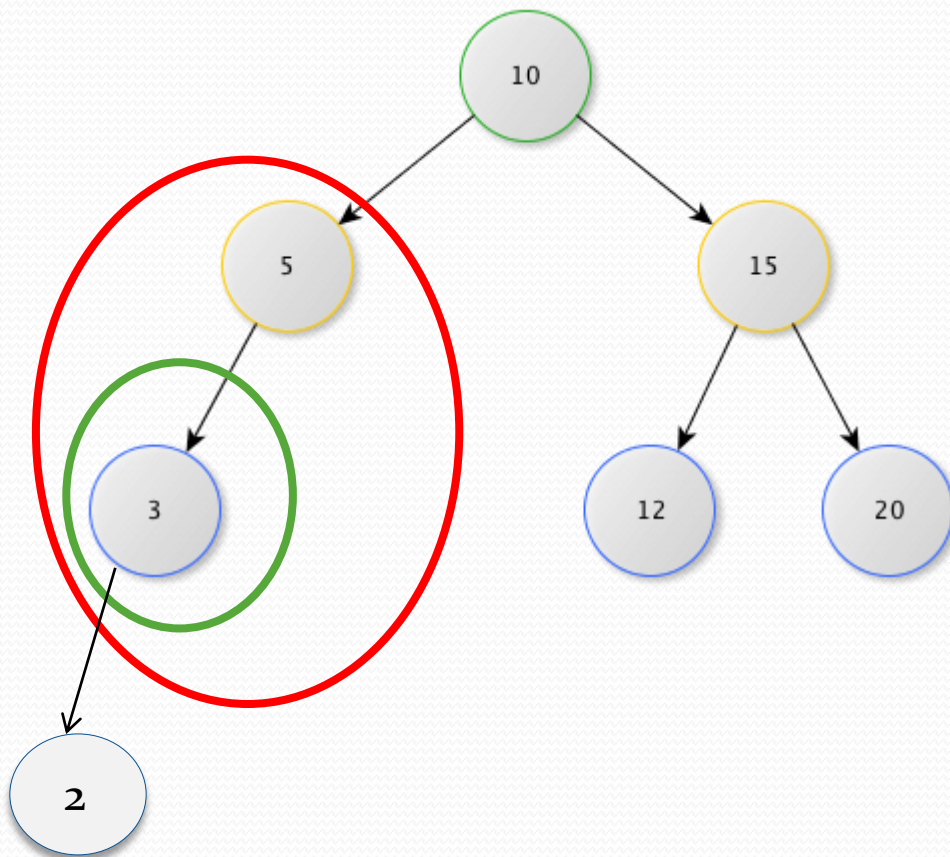
$2 < 10 \rightarrow SI$



```
Raíz=5  
Valor=2  
add(Raíz=5, Valor=2)
```



Insertar. Valor menor que la raíz



$2 < 5 \rightarrow \text{SI}$



**Raíz=3
Valor=2
add(Raíz=3, Valor=2)**

$2 < 3 \rightarrow \text{SI}$

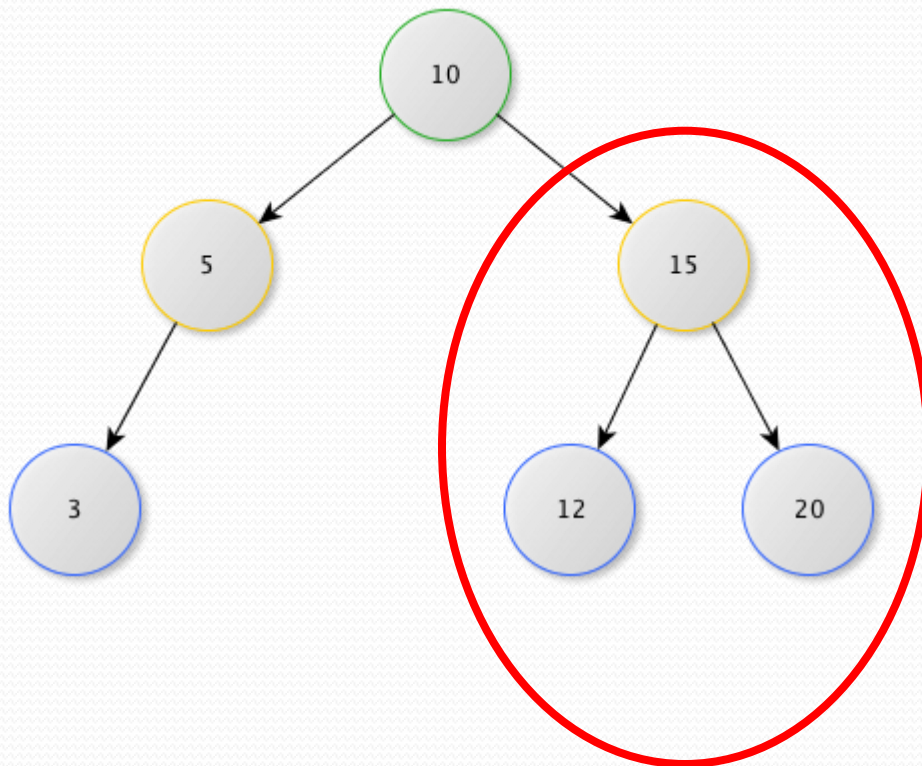
No tiene mas hijos $\rightarrow \text{SI}$



Inserta el 2 a la izquierda

Insertar. Valor mayor que la raíz

Insertar el nodo con valor 13



```
Raíz=10  
Valor=13  
add(Raíz=10, Valor=13)
```

$13 > 10 \rightarrow \text{SI}$



```
Raíz=15  
Valor=13  
add(Raíz=15, Valor=13)
```

Insertar. Valor mayor que la raíz

$13 > 15 \rightarrow \text{NO}$

$13 < 15 \rightarrow \text{SI}$



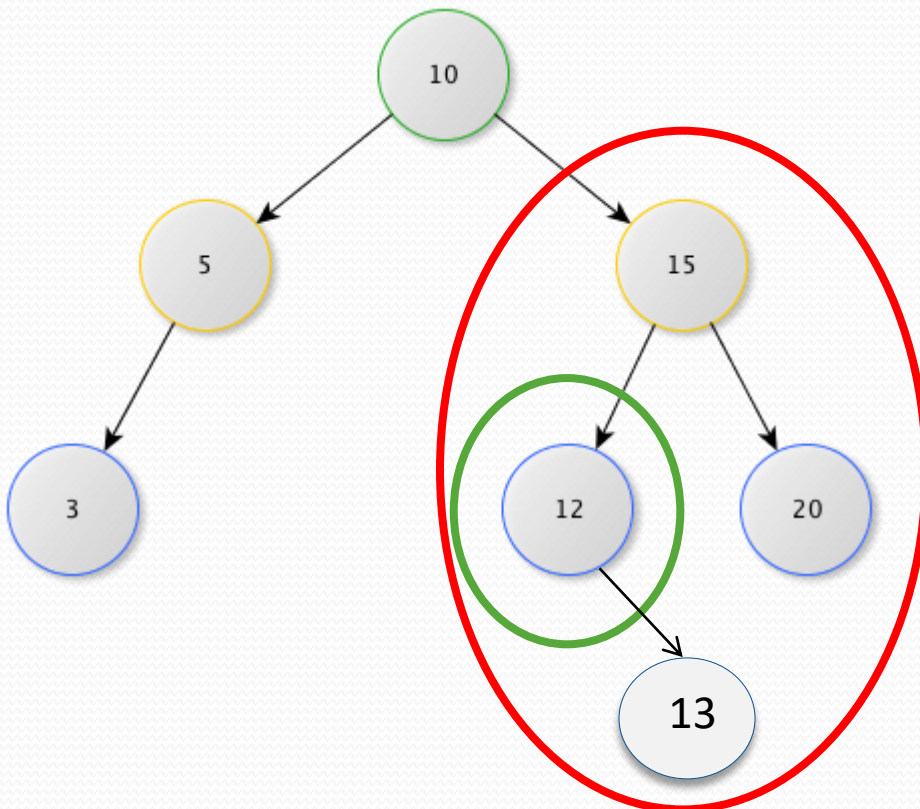
Raíz=12
Valor=13
add(Raíz=12, Valor=13)

$13 > 12 \rightarrow \text{SI}$

No tiene mas hijos



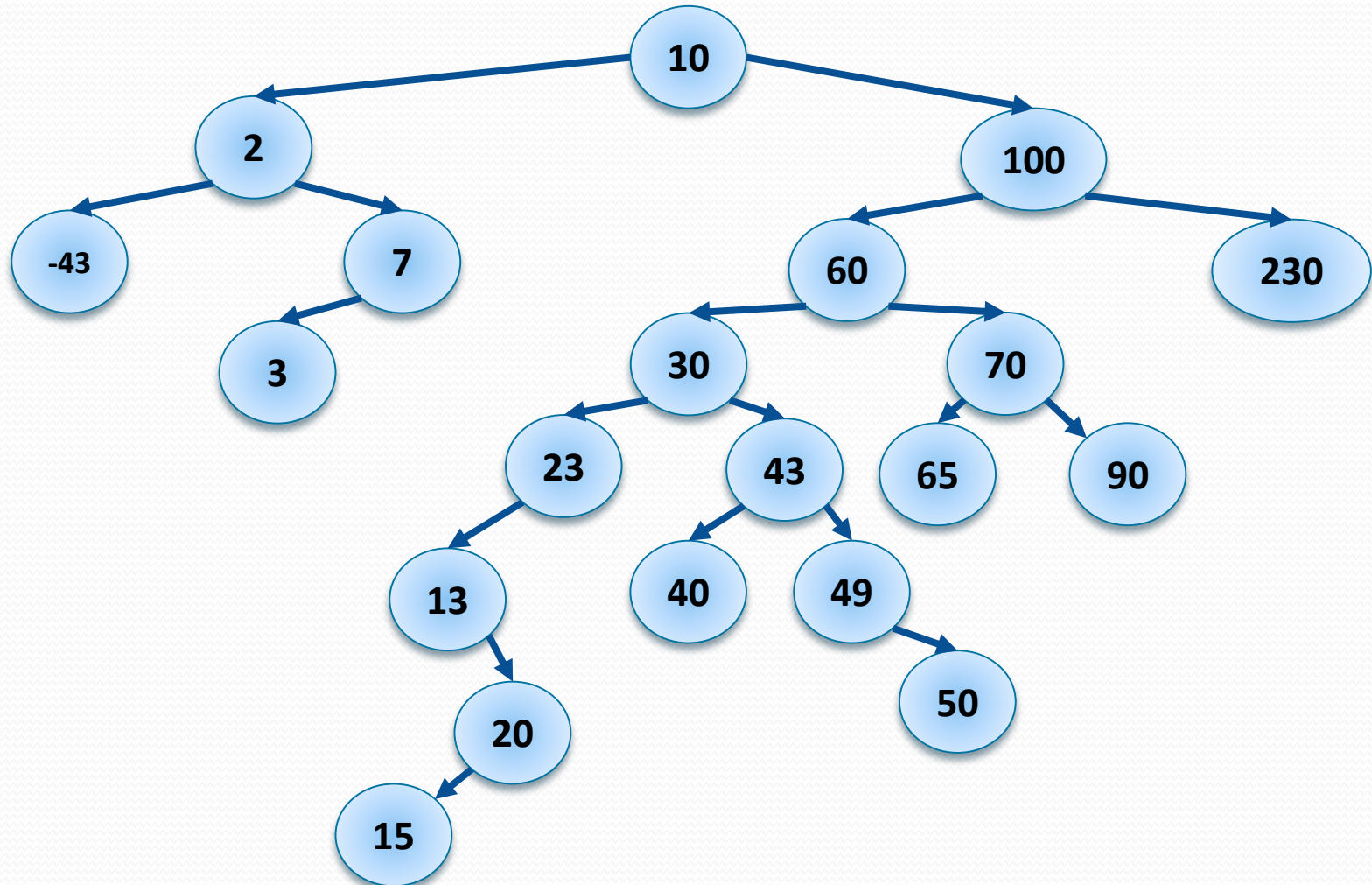
Inserta el 13 a la derecha



Insertar. Ejercicio

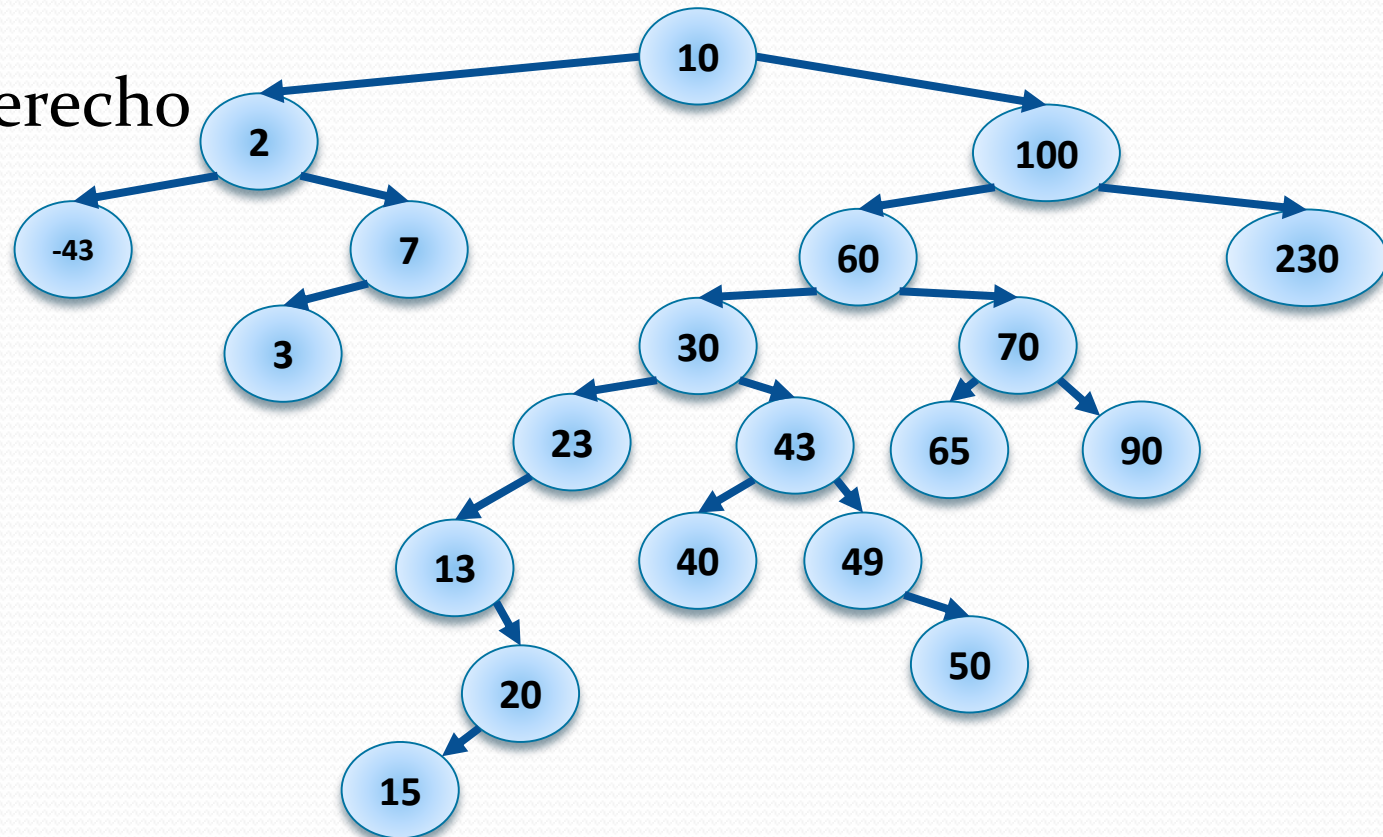
- Construye un árbol de búsqueda binario para una serie de nodos cuyas claves son:
 - 10, 100, 60, 30, 2, -43, 70, 90, 23, 43, 65, 13, 230, 49, 7, 40, 50, 20, 15, 3
- Ten en cuenta que los nodos han aparecido de uno en uno en orden, siendo el primero de ellos el nodo con clave 10 y el último, el nodo con clave 3

Insertar. Ejercicio



Recorridos. InOrder

- Subárbol izquierdo
- Raíz
- Subárbol derecho



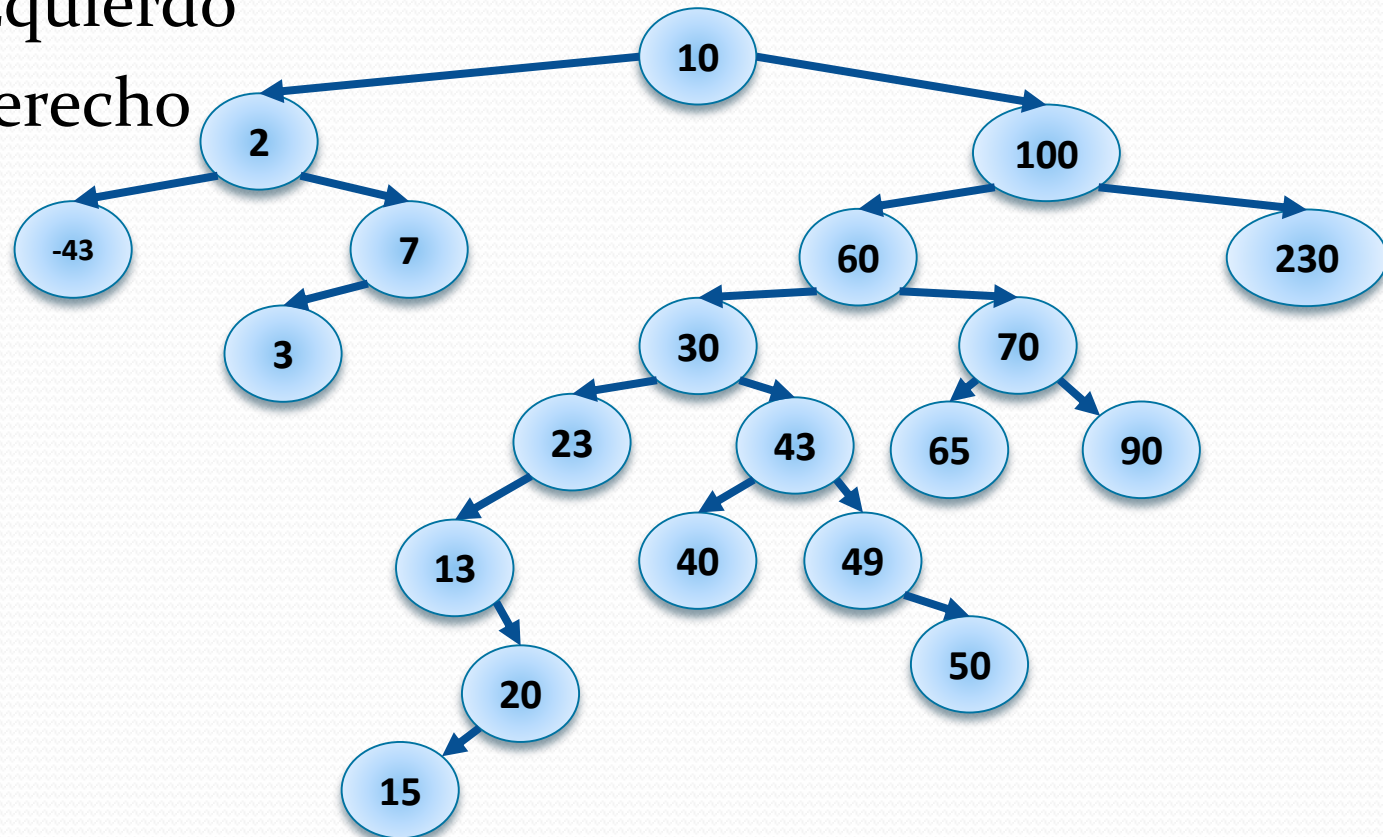
Recorridos. InOrder

- Subárbol izquierdo
- Raíz
- Subárbol derecho

-43, 2, 3, 7, 10, 13, 15, 20, 23, 30, 40,
43, 49, 50, 60, 65, 70, 90, 100, 230

Recorridos. PreOrder

- Raíz
- Subárbol izquierdo
- Subárbol derecho



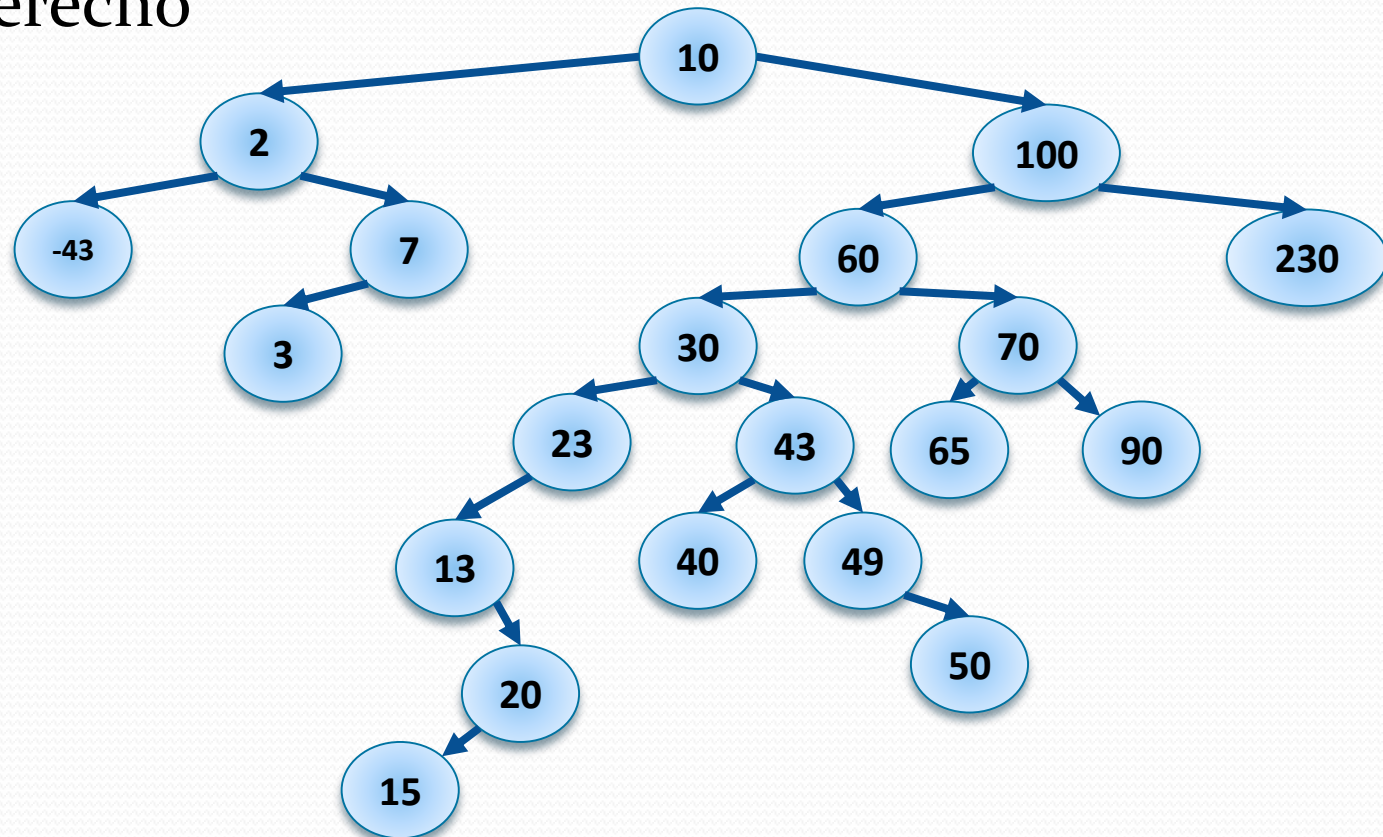
Recorridos. PreOrder

- Raíz
- Subárbol izquierdo
- Subárbol derecho

10, 2, -43, 7, 3, 100, 60, 30, 23, 13, 20,
15, 43, 40, 49, 50, 70, 65, 90, 230

Recorridos. PostOrder

- Subárbol izquierdo
- Subárbol derecho
- Raíz



Recorridos. PostOrder

- Subárbol izquierdo
- Subárbol derecho
- Raíz

-43, 3, 7, 2, 15, 20, 13, 23, 40, 50, 49,
43, 30, 65, 90, 70, 60, 230, 100, 10

Borrar un nodo

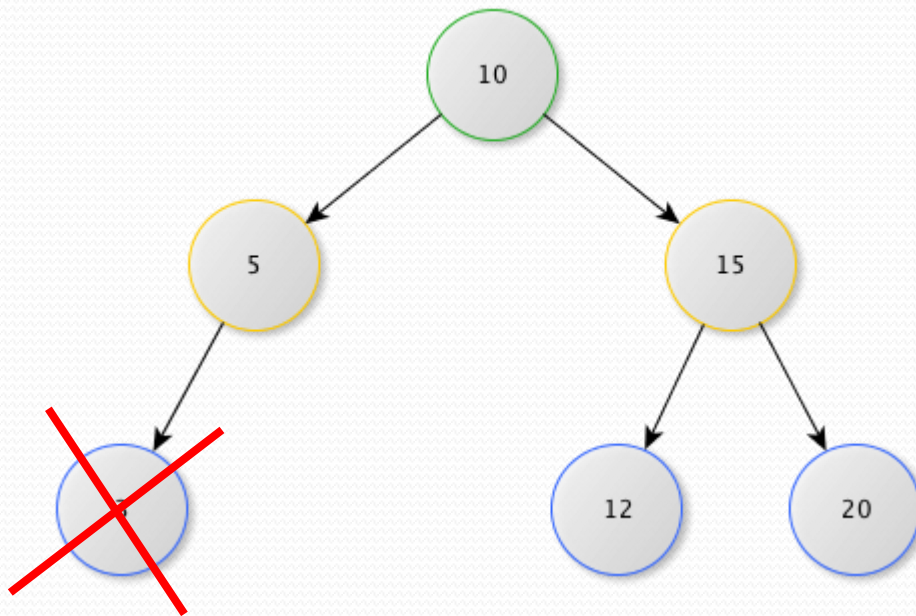
- Cuatro casos

- El nodo no existe
- El nodo no tiene hijos
- El nodo tiene un solo hijo
- El nodo tiene dos hijos



Procedimiento
recursivo

Borrar. Nodo sin hijos



Borrar el 3

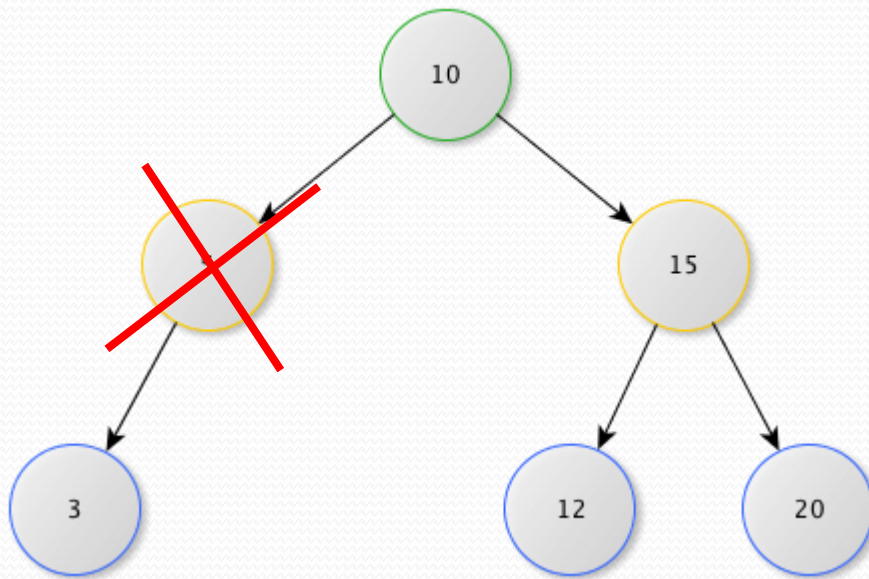


Hijo Izdo del 3 \rightarrow null
Hijo Dcho del 3 \rightarrow null
Padre del 3 \rightarrow 5



Como 3 es el hijo izquierdo del 5
pongo el hijo izquierdo del 5 a null

Borrar. Nodo con un hijo



Borrar el 5

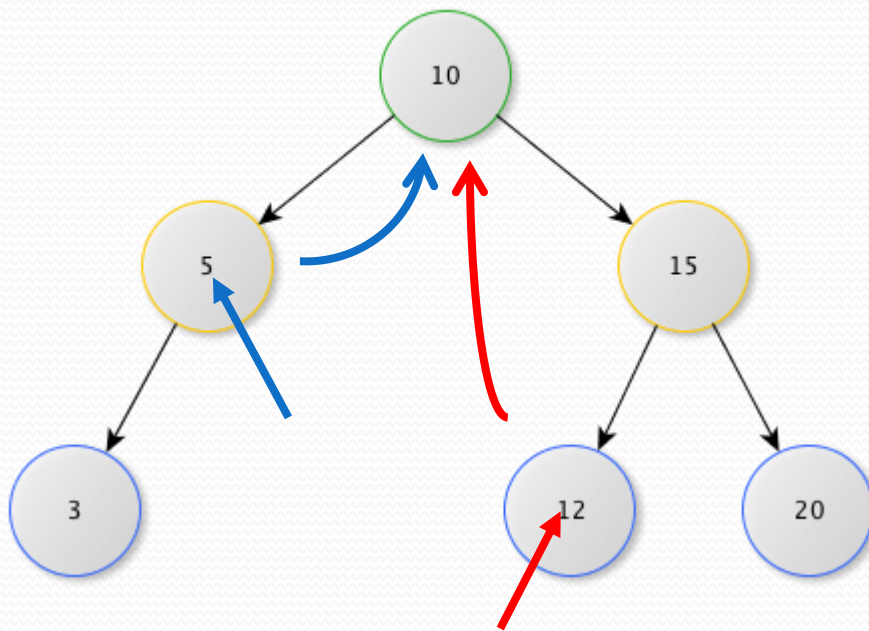


Hijo Izdo del 5 \rightarrow 3
Hijo Dcho del 5 \rightarrow null
Padre del 5 \rightarrow 10



Como 5 es el hijo izquierdo del 10
asigno como nuevo hijo izquierdo el
hijo del nodo a borrar

Borrar. Nodo con dos hijos



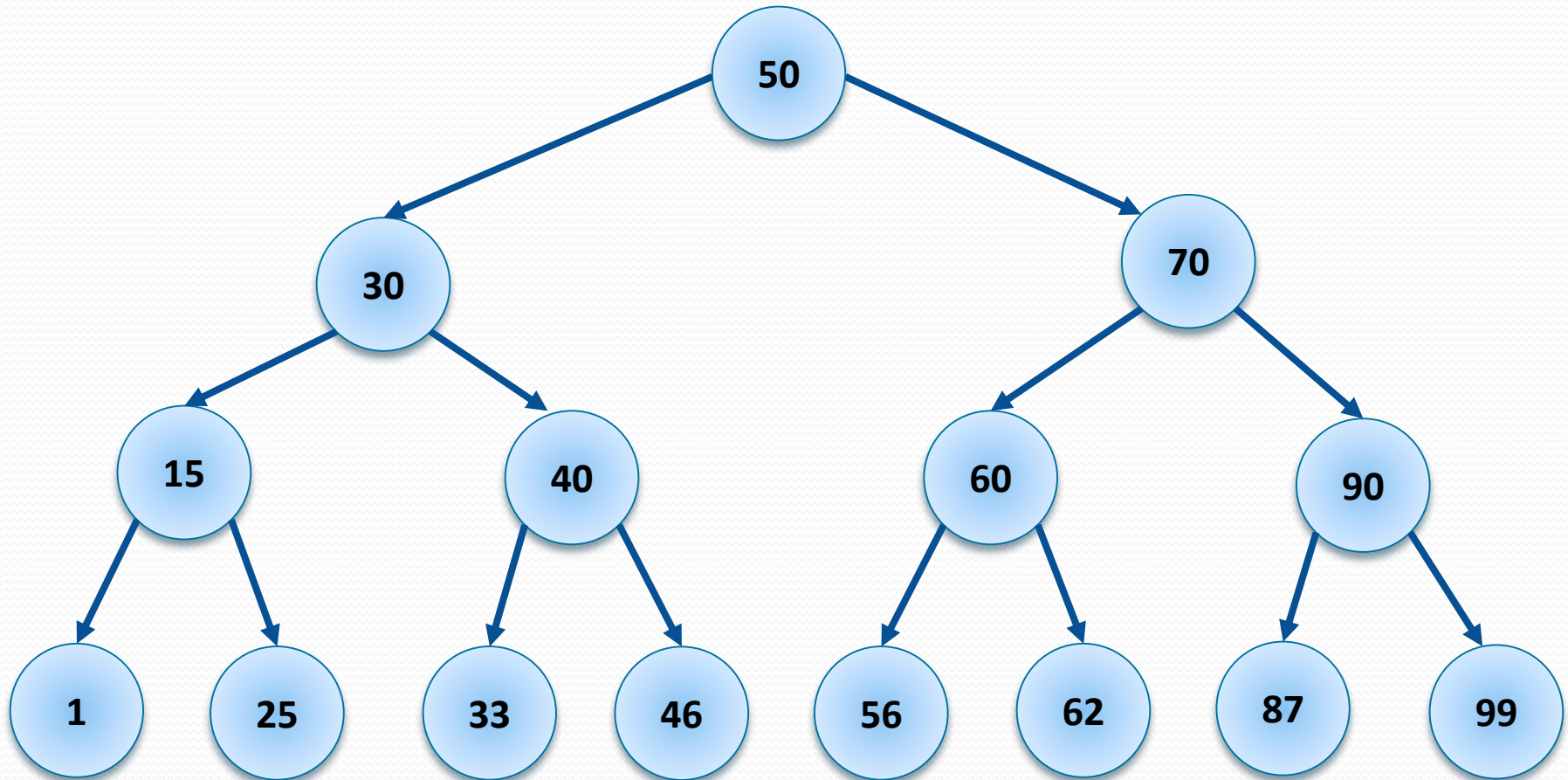
Borrar el 10

Hijo Izdo del 10 \rightarrow 5
Hijo Dcho del 10 \rightarrow 15
Padre del 15 \rightarrow ---

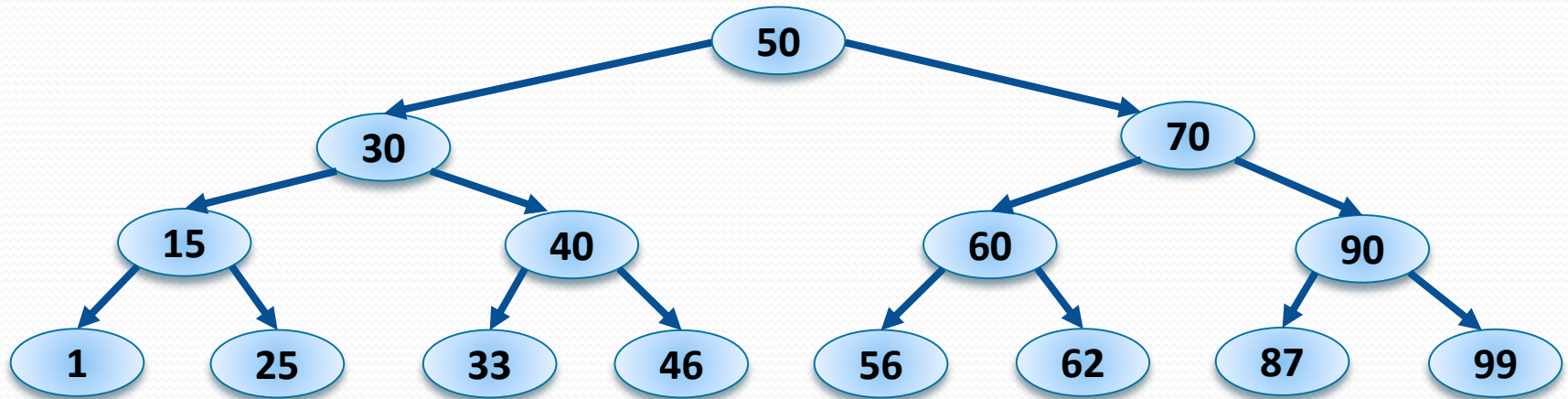


Busco el mayor del subárbol izquierdo
Busco el menor del subárbol derecho

Borrar: Ejercicio



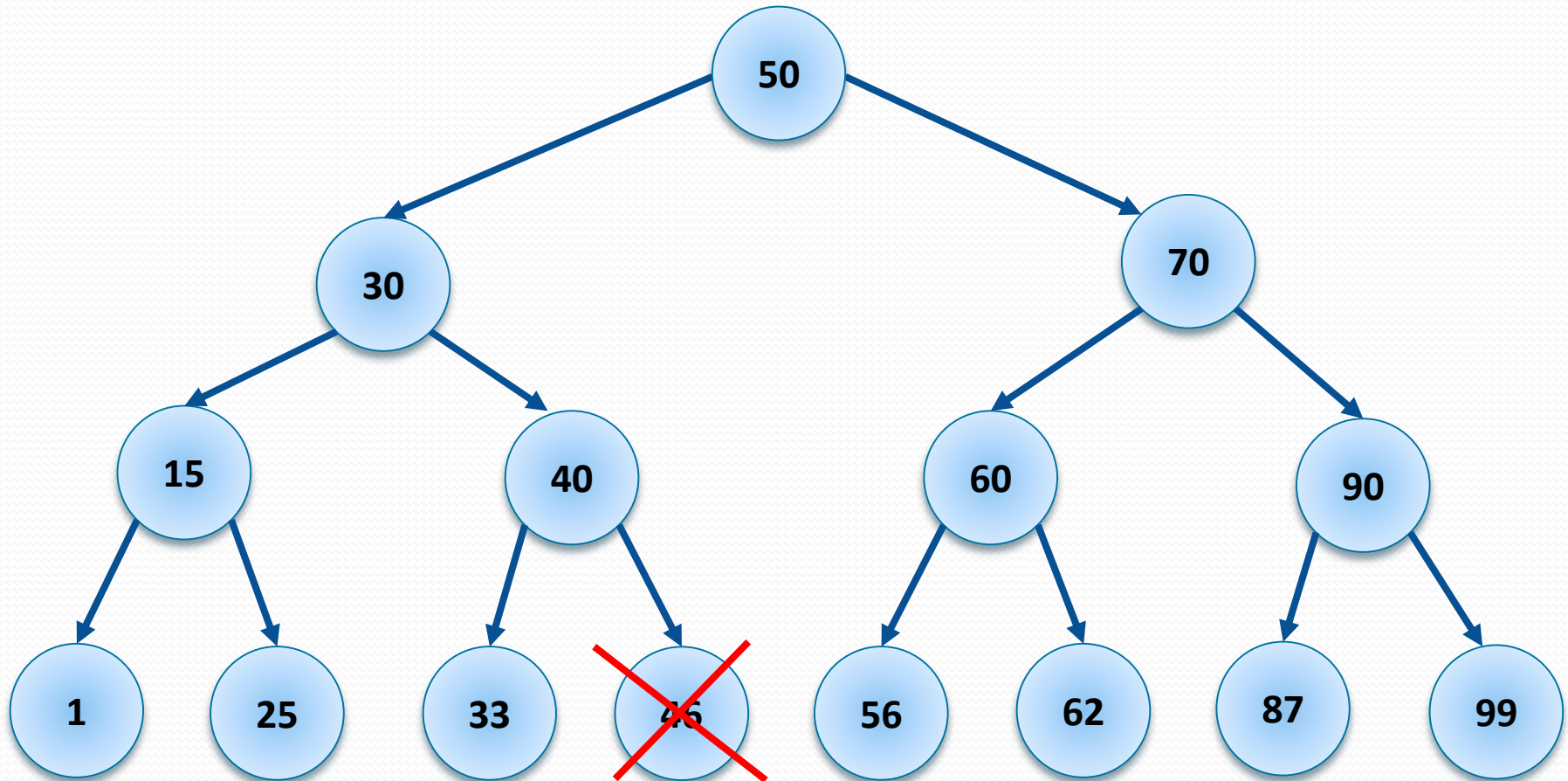
Borrar: Ejercicio



Borrar y en este orden:
46, 40, 15, 30, 70, 60, 87, 90, 50

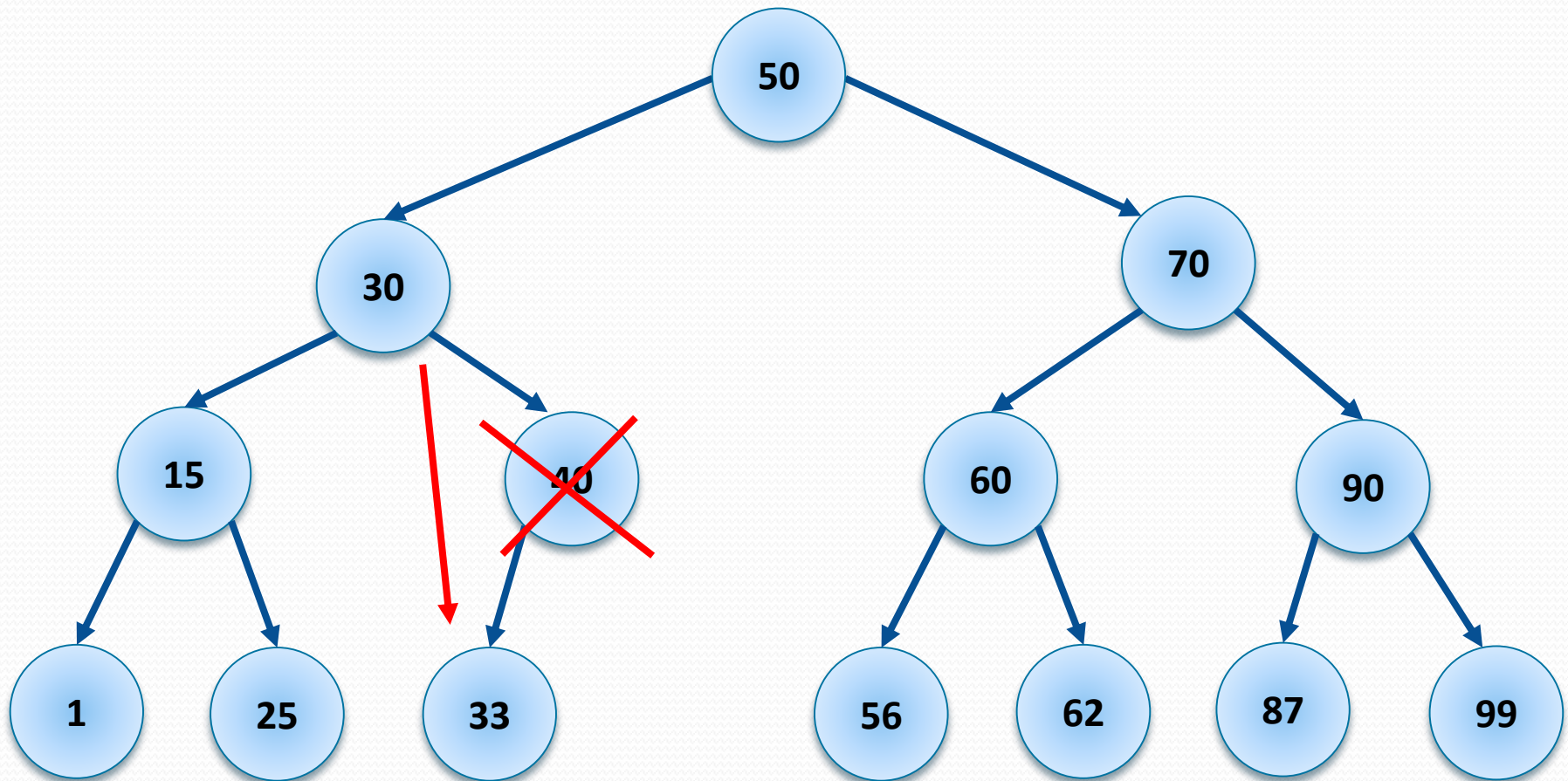
Borrar: Ejercicio

Borrar el 46



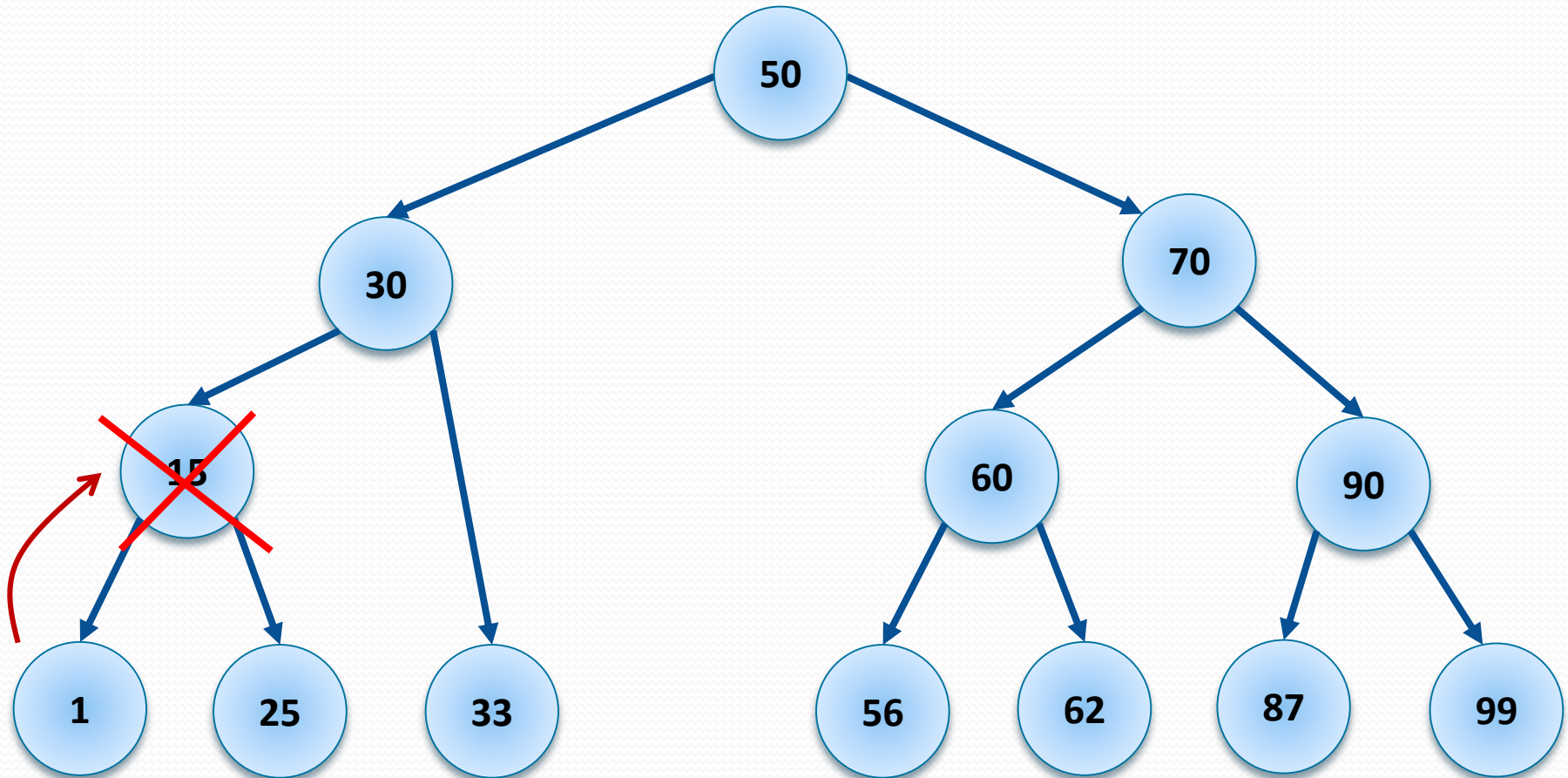
Borrar: Ejercicio

Borrar el 40



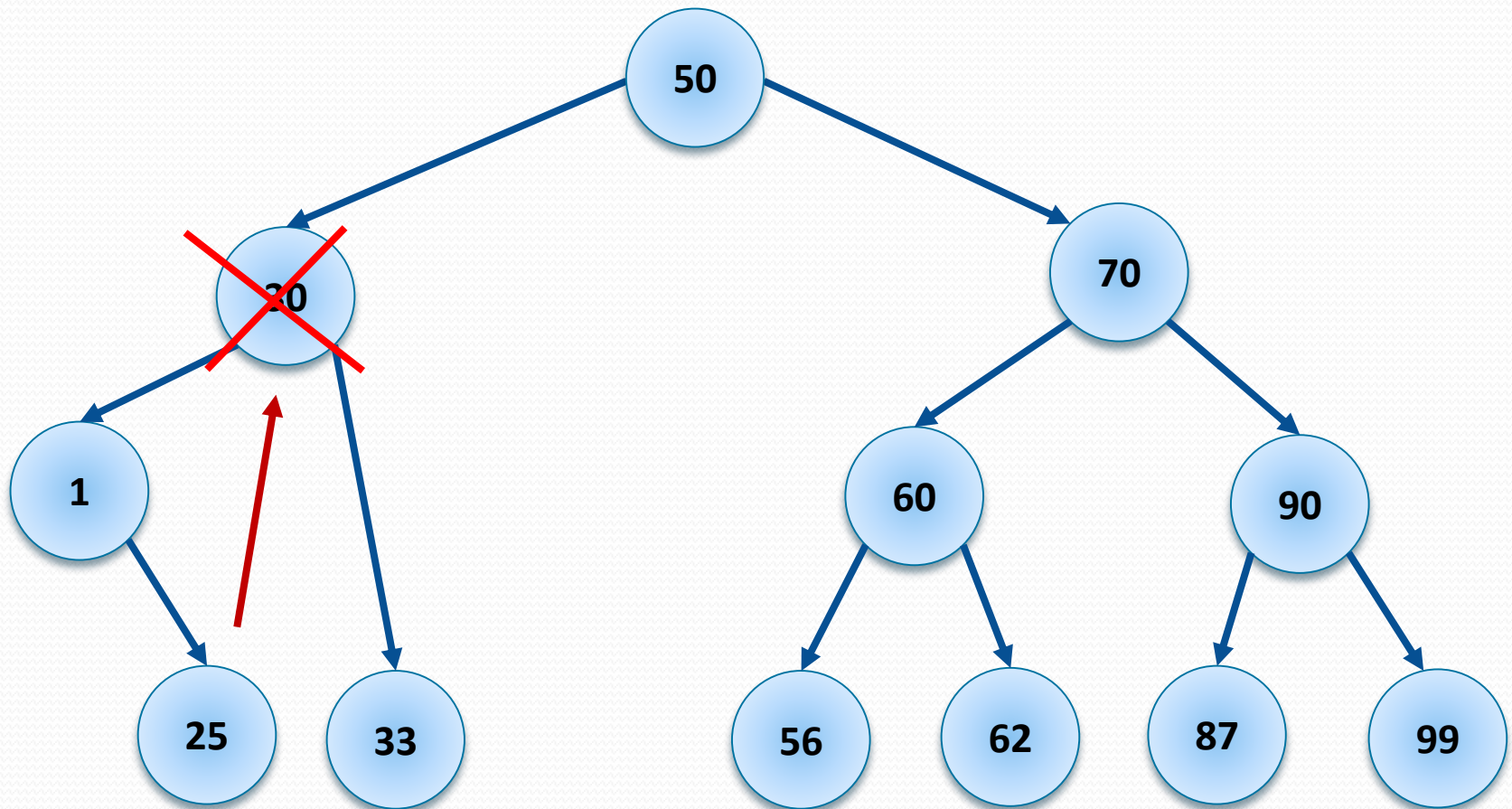
Borrar: Ejercicio

Borrar el 15



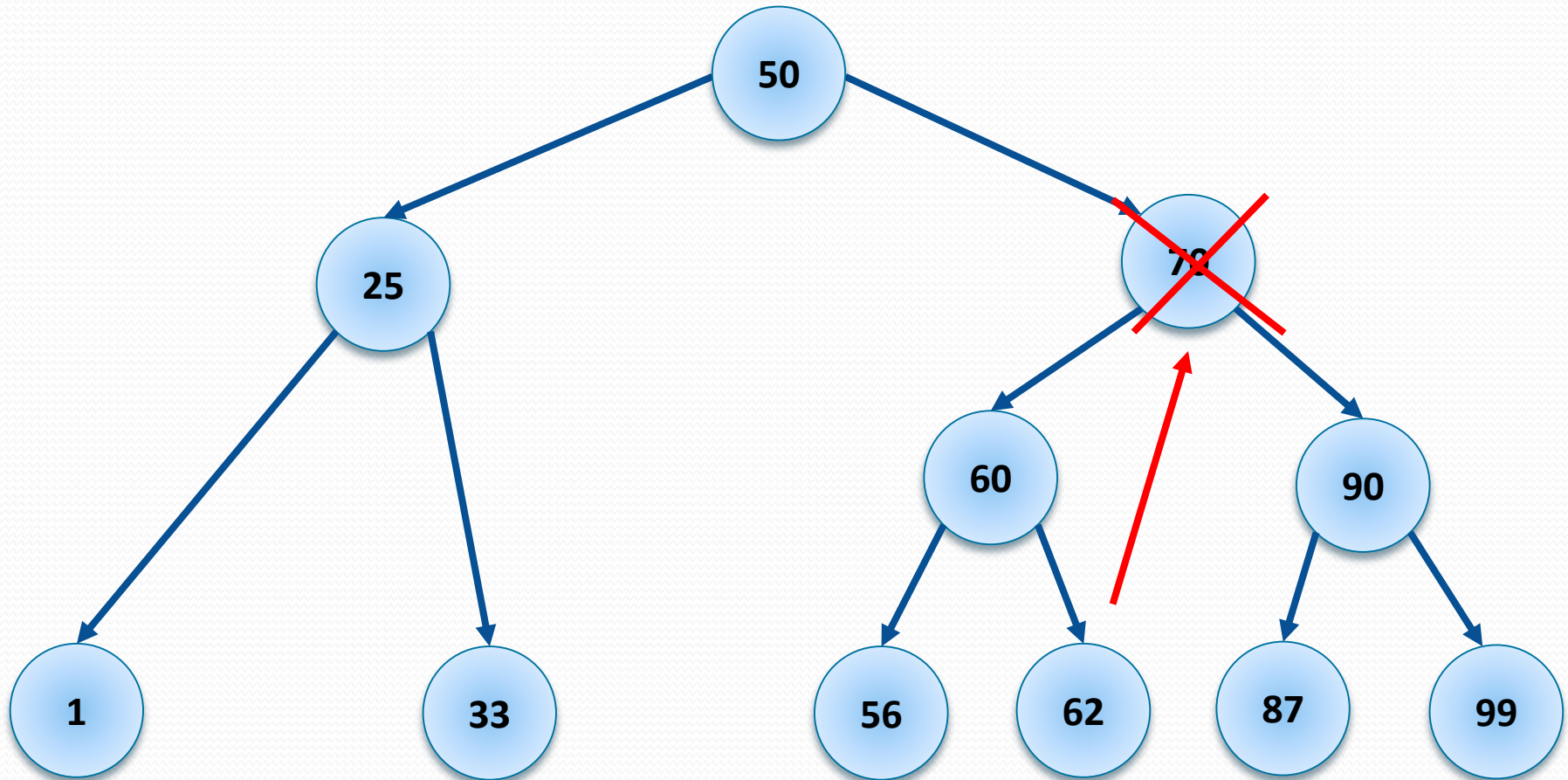
Borrar: Ejercicio

Borrar el 30



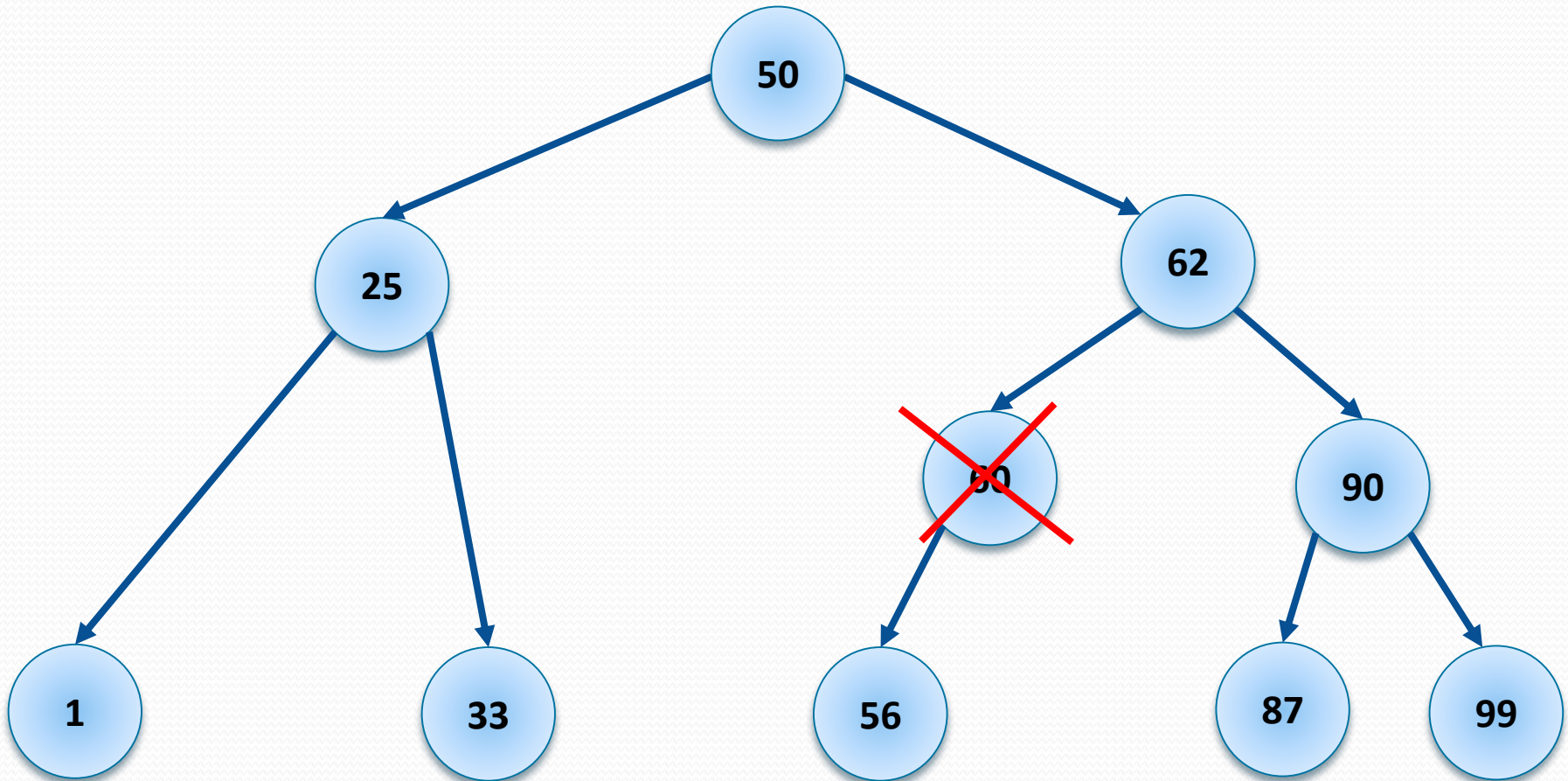
Borrar: Ejercicio

Borrar el 70



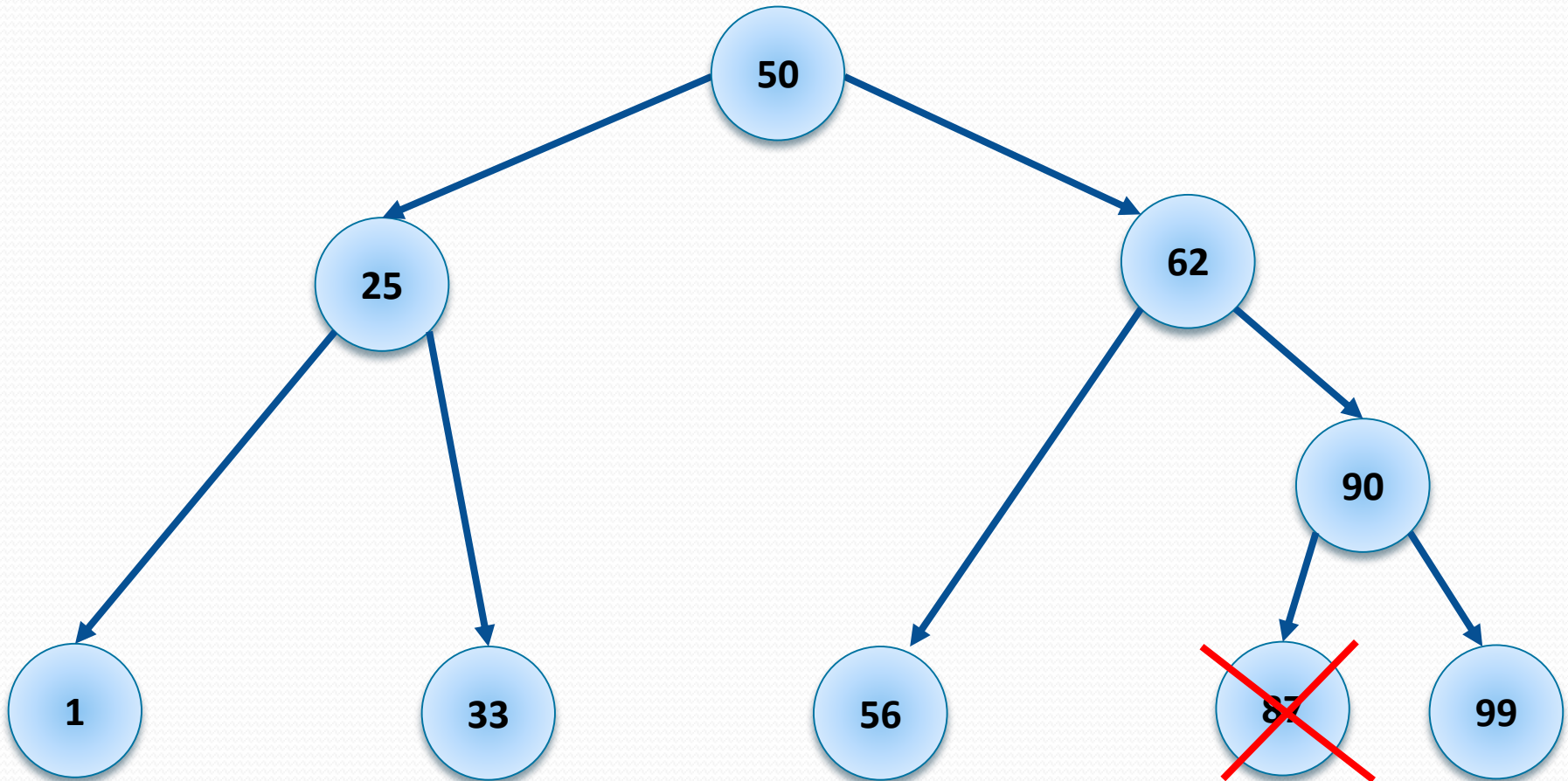
Borrar: Ejercicio

Borrar el 60



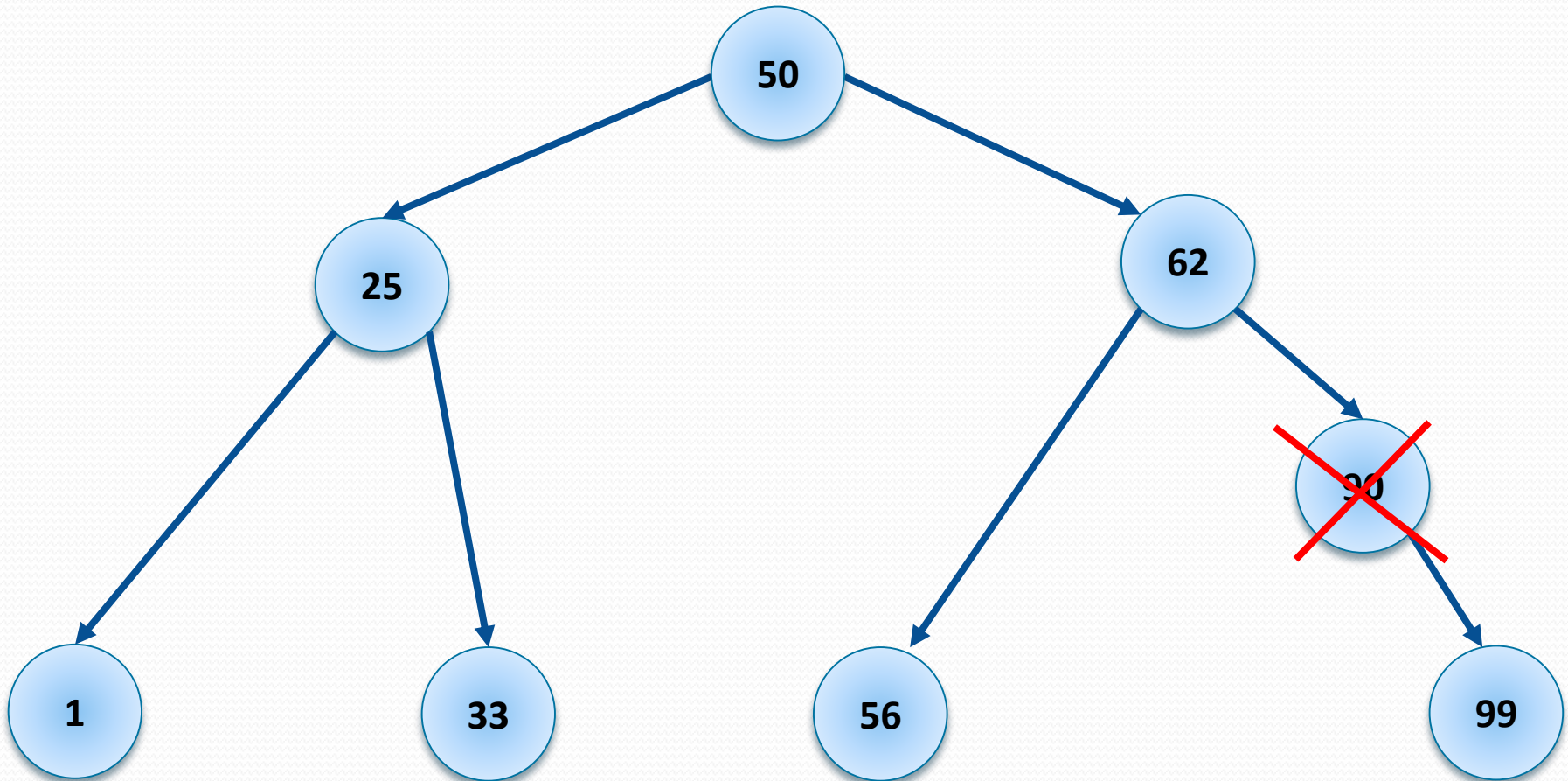
Borrar: Ejercicio

Borrar el 87



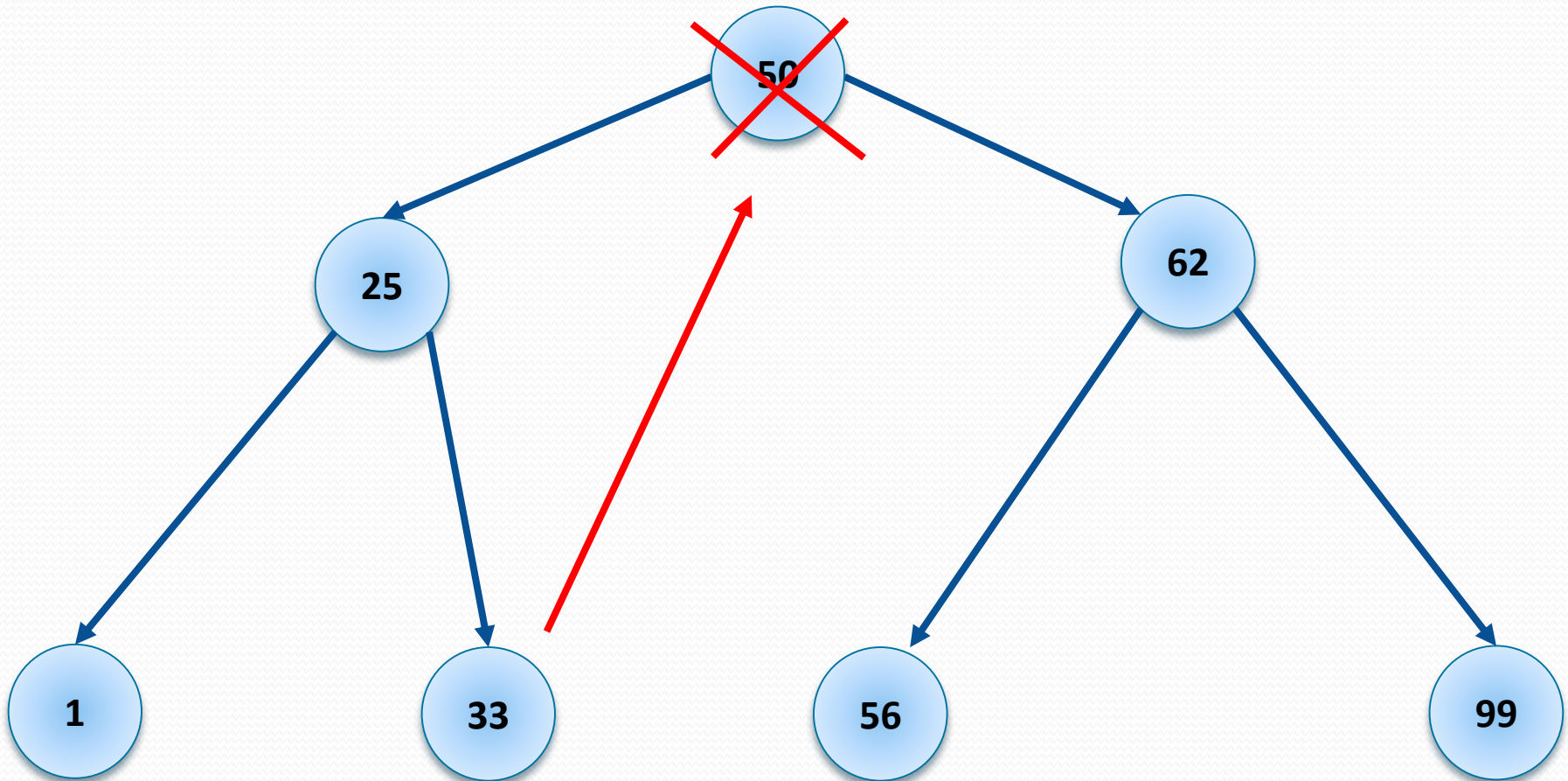
Borrar: Ejercicio

Borrar el 90

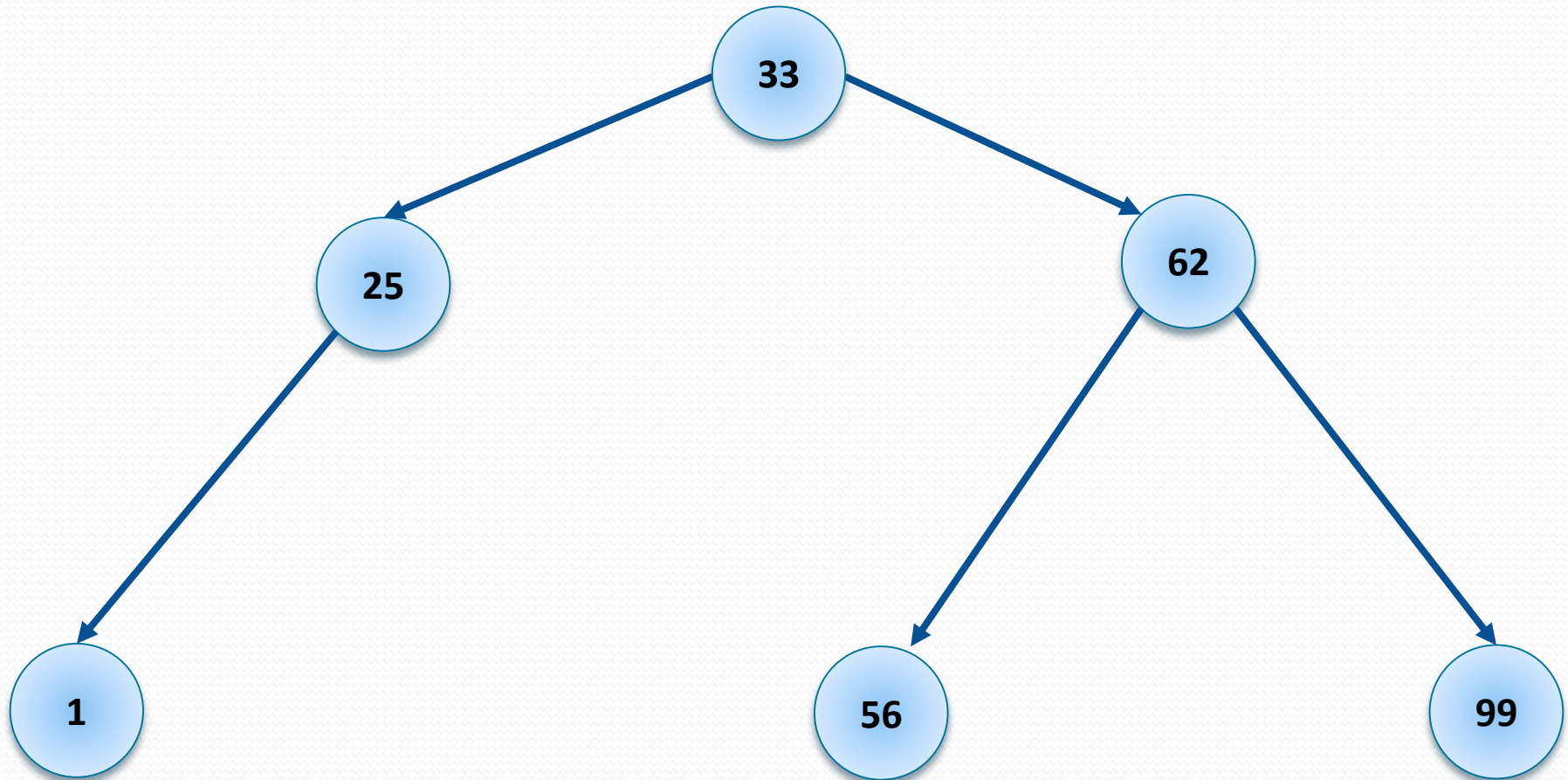


Borrar: Ejercicio

Borrar el 50



Borrar: Ejercicio



Ejercicio

- Construye un árbol de búsqueda binario para una serie de nodos cuyas claves son:
 - 50, 25, 75, 18, 28, 100, 32, 80, 30, 40, 29, 31, 90, 95
- Muestra el recorrido inOrden, preOrden y postOrden
- Borra las siguiente claves:
 - 32, 100, 50, 80, 40, 28
- Muestra el recorrido inOrden, preOrden y postOrden