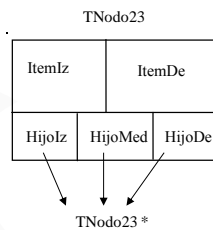


### 3.3. Árboles 2-3

#### DEFINICIONES

- Un árbol 2-3 es un árbol que está vacío o satisface las siguientes propiedades:
  - Los nodos pueden tener 2 ó 3 hijos (2-nodo ó 3-nodo)
  - Cumple las propiedades de árbol multcamino de búsqueda
  - Todas las hojas están en el mismo nivel
- Representación



```
class TArb23 {
    public:
        .....
    private:
        TNode23 * farb;
};
```

1

### 3.3. Árboles 2-3

#### OPERACIONES BÁSICAS. PROPIEDADES

- Operaciones básicas:
  - Búsqueda (similar a los árboles multcamino de búsqueda)
  - Inserción (se realiza en las hojas. Se pueden producir reestructuraciones del árbol en el camino de vuelta)
  - Borrado (se realiza en las hojas. Se pueden producir reestructuraciones del árbol en el camino de vuelta)
- Propiedades:
  - En un árbol 2-3 de altura  $h$  tenemos:
    - $2^h - 1$  elementos si todos los nodos son del tipo 2-nodo
    - $3^h - 1$  elementos si todos los nodos son del tipo 3-nodo
 por lo que la altura de un árbol 2-3 con  $n$  elementos se encuentra entre los límites:  $\log_3(n+1)$  y  $\log_2(n+1)$
  - Las reestructuraciones se realizan desde las hojas hacia la raíz

2

### 3.3. Árboles 2-3

#### OPERACIONES BÁSICAS. INSERCIÓN (I)

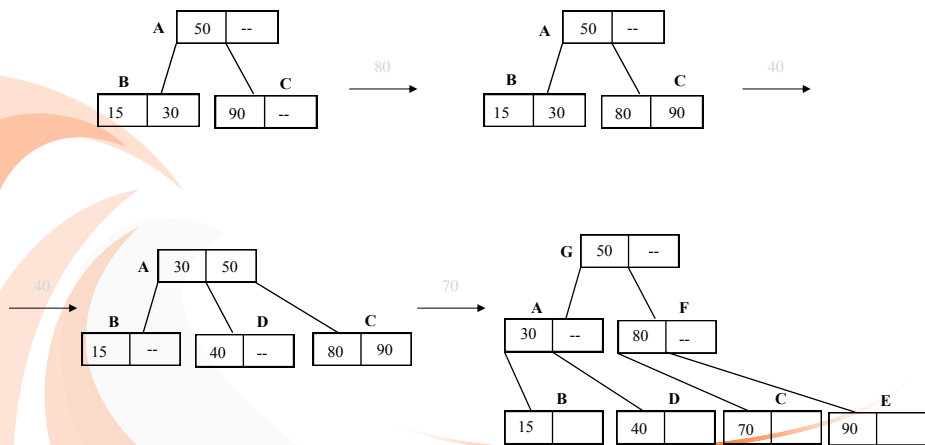
- Pueden ocurrir dos casos:
  - El elemento a insertar irá en un 2-nodo: INSERCIÓN DIRECTA
  - El elemento a insertar irá en un 3-nodo: HAY QUE CREAR UN NUEVO NODO

3

### 3.3. Árboles 2-3

#### OPERACIONES BÁSICAS. INSERCIÓN (II)

- Ejemplo: Insertar en el siguiente árbol 2-3 los elementos: 80, 40 y 70



4

### 3.3. Árboles 2-3

#### EJERCICIOS inserción

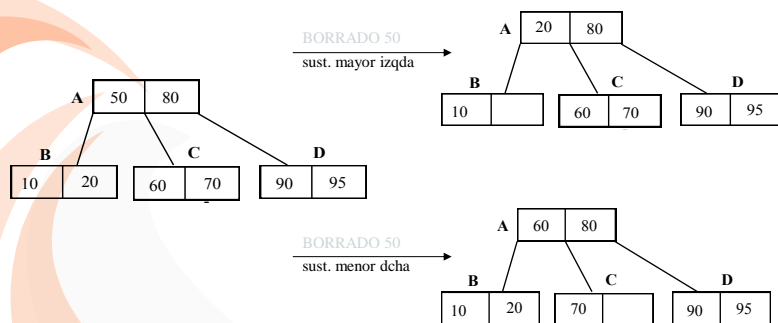
- 1) En el árbol 2-3 obtenido anteriormente, insertar los elementos 45, 47, 35, 33, 48, 49, 43 y 42

5

### 3.3. Árboles 2-3

#### OPERACIONES BÁSICAS. BORRADO (I)

- Pueden ocurrir dos casos:
  - El elemento a borrar está en un 3-nodo: BORRADO DIRECTO
  - El elemento a borrar irá en un 2-nodo: realizar una COMBINACIÓN o ROTACIÓN

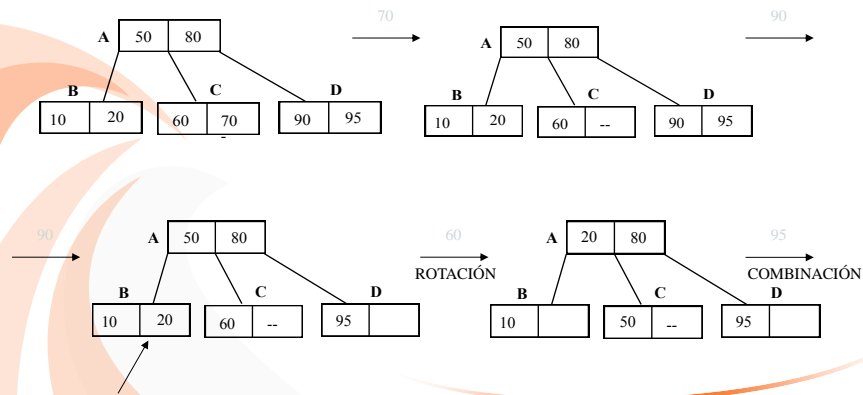


6

### 3.3. Árboles 2-3

#### OPERACIONES BÁSICAS. BORRADO (II)

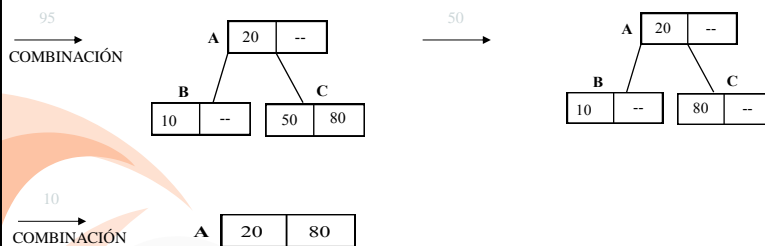
- Ejemplo: Borrar en el siguiente árbol 2-3 los elementos: 70, 90, 60, 95, 50 y 10. (Criterios: (1) si el nodo tiene dos hijos hay que sustituir por el mayor de la izquierda, (2) si el 2-nodo tiene dos hermanos, consultar el hermano de la izquierda)



7

### 3.3. Árboles 2-3

#### OPERACIONES BÁSICAS. BORRADO (III)

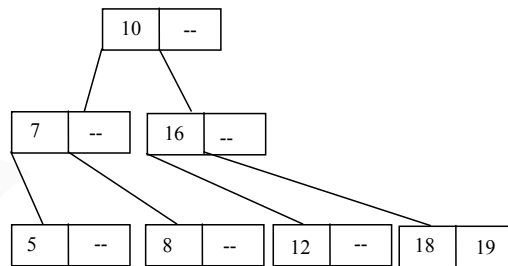


8

### 3.3. Árboles 2-3

#### EJERCICIOS borrado

- 1) Dado el siguiente árbol 2-3 borrar los elementos 10, 7 y 18. (Criterios: (1) si el nodo tiene dos hijos hay que sustituir por el menor de la derecha, (2) si el 2-nodo tiene dos hermanos, consultar el hermano de la derecha)

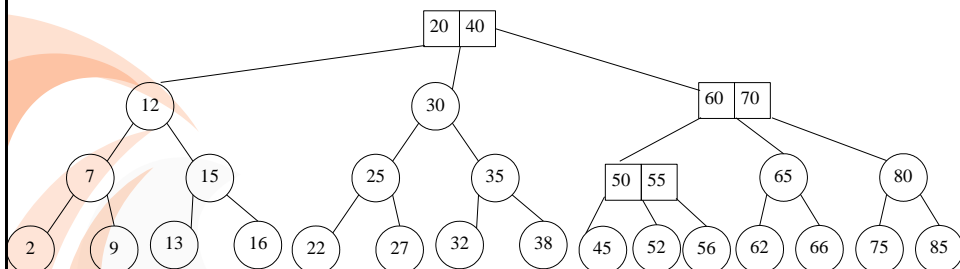


9

### 3.3. Árboles 2-3

#### EJERCICIOS borrado

- 2) Dado el siguiente árbol 2-3 borrar los elementos 20, 30, 70 y 12. (Criterios: (1) si el nodo tiene dos hijos hay que sustituir por el menor de la derecha, (2) si el 2-nodo tiene dos hermanos, consultar el hermano de la derecha)



10

### 3.3. Árboles 2-3

Preguntas de tipo test: Verdadero vs. Falso

- Un árbol 2-3 es un árbol 2-camino de búsqueda
- El número mínimo de elementos que se pueden almacenar en un árbol 2-3 de altura  $h$  es  $3^h - 1$
- El grado del árbol 2-3 es 2