Grado en Ingeniería Información

Estructura de Datos y Algoritmos

Sesión 10

Curso 2022-2023

Marta N. Gómez

UNIVERSIDAD NEBRIJA

T3. Tipos Abstractos de Datos (TAD)

- **Árboles.**
  - Conceptos generales
  - Realización del TAD Árbol Binario
  - Recorridos de Árboles Binarios
  - Árboles Binarios de Búsqueda (ABB)
  - **Árboles Equilibrados (AVL)**
  - Montículos

UNIVERSIDAD NEBRIJA

ÍNDICE

Un **Árbol Equilibrado es un árbol binario de búsqueda (ABB)** con restricciones estructurales que garantizan que su altura sea logarítmica (h ~ O(log n)).

Hay varios tipos de árboles equilibrados:

– Árboles AVL.

– Árboles rojo-negro.

– etc.

Un **árbol Equilibrado o Balanceado o AVL** (matemáticos rusos Adelson-Velskii y Landis) es un **ABB** donde para cada nodo las **alturas de sus subárboles, izquierdo y derecho, no difieren en más de 1**.

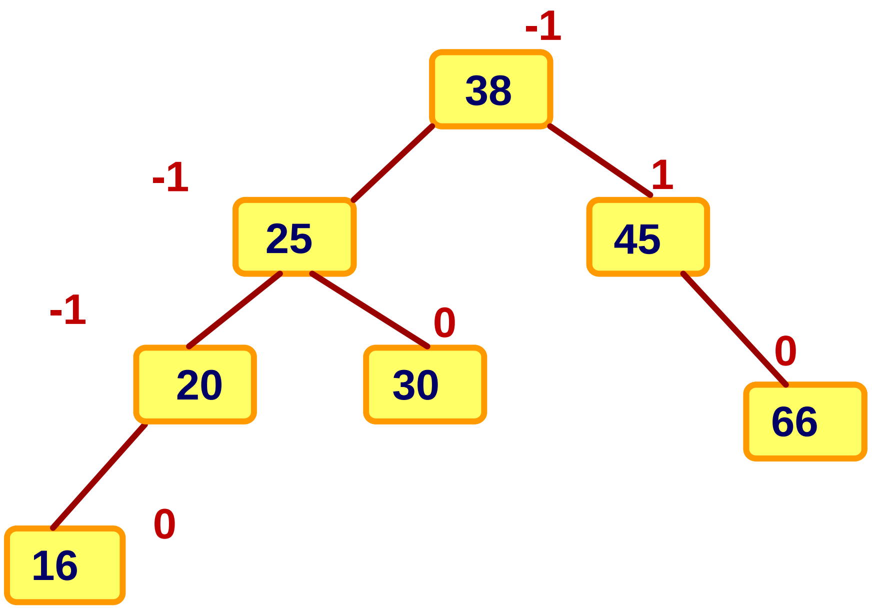
OPERACIONES EN ÁRBOLES AVL: Los AVL son también ABB de modo que conservan todas las operaciones de éstos. Las nuevas operaciones son las **rotaciones** para **equilibrar el árbol**, que se hacen como parte de las **operaciones de inserción** y **borrado**.

FACTOR DE EQUILIBRIO:

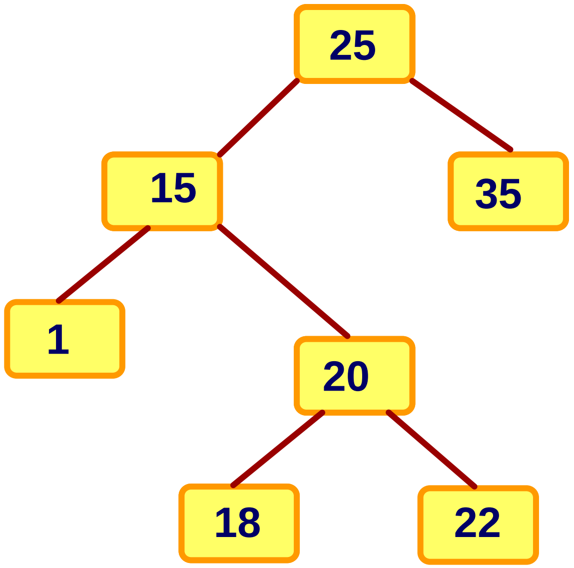
El **factor de equilibrio (FE)** es la **diferencia entre las alturas del subárbol derecho e izquierdo**:

FE = altura subárbol derecho – altura subárbol izquierdo

Por definición, el valor del FE en un árbol AVL debe ser **-1**, **0** ó **1**.



-1
38
-1
25
1
45
-1
20
0
30
0
66
16
0



25
15
35
1
20
18
22

Insertar un nuevo nodo hoja en un AVL

1. Las ramas izquierda (Ri) y la derecha (Rd) del árbol tienen la misma altura (**$h_{Ri} = h_{Rd}$**): **NO se rompe el equilibrio.**
2. Las ramas izquierda y derecha del árbol tienen altura diferente:
   a) Suponiendo que **$h_{Ri} < h_{Rd}$**, puede ocurrir:
      - Si se inserta el nodo en la **Ri** entonces **$h_{Ri} = h_{Rd}$** y se mejora el equilibrio.

   b) Si se inserta el nodo en la **Rd** entonces se rompe el criterio de equilibrio del árbol y **h_{Rd}**, puede ocurrir:
      - Si se inserta el nodo es necesario **reestructurarlo**.

   c) Suponiendo que **h_{Ri} >** en la **Rd** entonces **$h_{Ri} = h_{Rd}$** y se mejora el equilibrio.
      - Si se inserta el nodo en la **Ri** entonces se rompe el criterio de equilibrio del árbol y es necesario **reestructurarlo.**

La **reorganización** o **reestructuración** que se debe realizar en el árbol es a través de las siguientes **rotaciones**:

- ***Rotaciones simples***, afectan a **dos nodos**.

  Una vez realizada la rotación los factores de equilibrio de los nodos implicados será siempre 0.

- ***Rotaciones dobles***, afecta a **tres nodos**.

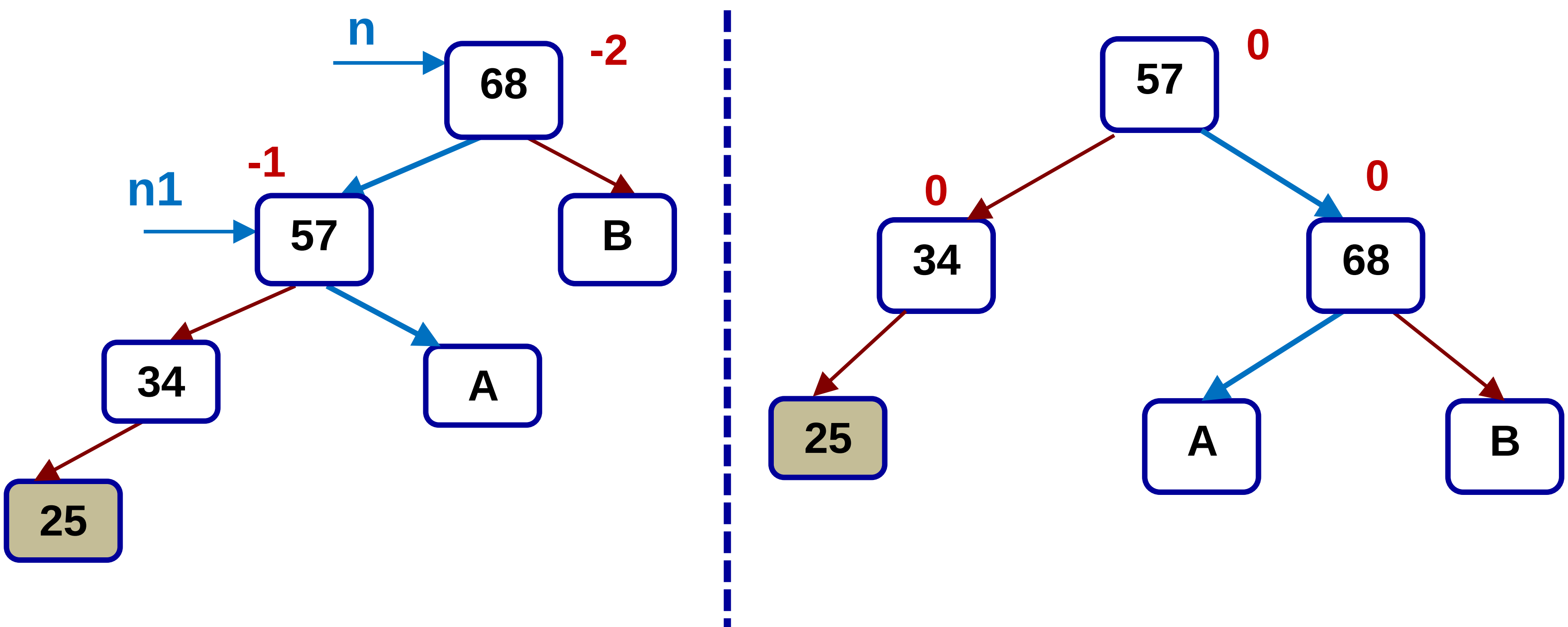
Rotación simple a la izquierda o rotación Izq-Izq:

Se realiza cuando el **factor de equilibrio de un nodo es -2**, es decir, su subárbol izquierdo es dos unidades más alto que el derecho, y, además, **la raíz del subárbol izquierdo tiene un FE de -1**, es decir, el **árbol está cargado a la izquierda**.

Rotación simple Izquierda-Izquierda:

1. Pasar subárbol derecho del nodo **n1** como subárbol izquierdo del nodo **n**.
2. El árbol **n** pasa a ser el subárbol derecho del nodo **n1**.
3. El nodo **n1** toma la posición del nodo **n** (la entrada al árbol es ahora el nodo **n1**).

```
n->hizq = n1->hdch;
n1->hdch = n
n = n1
```



Rotación simple a la derecha o rotación Dcha-Dcha:

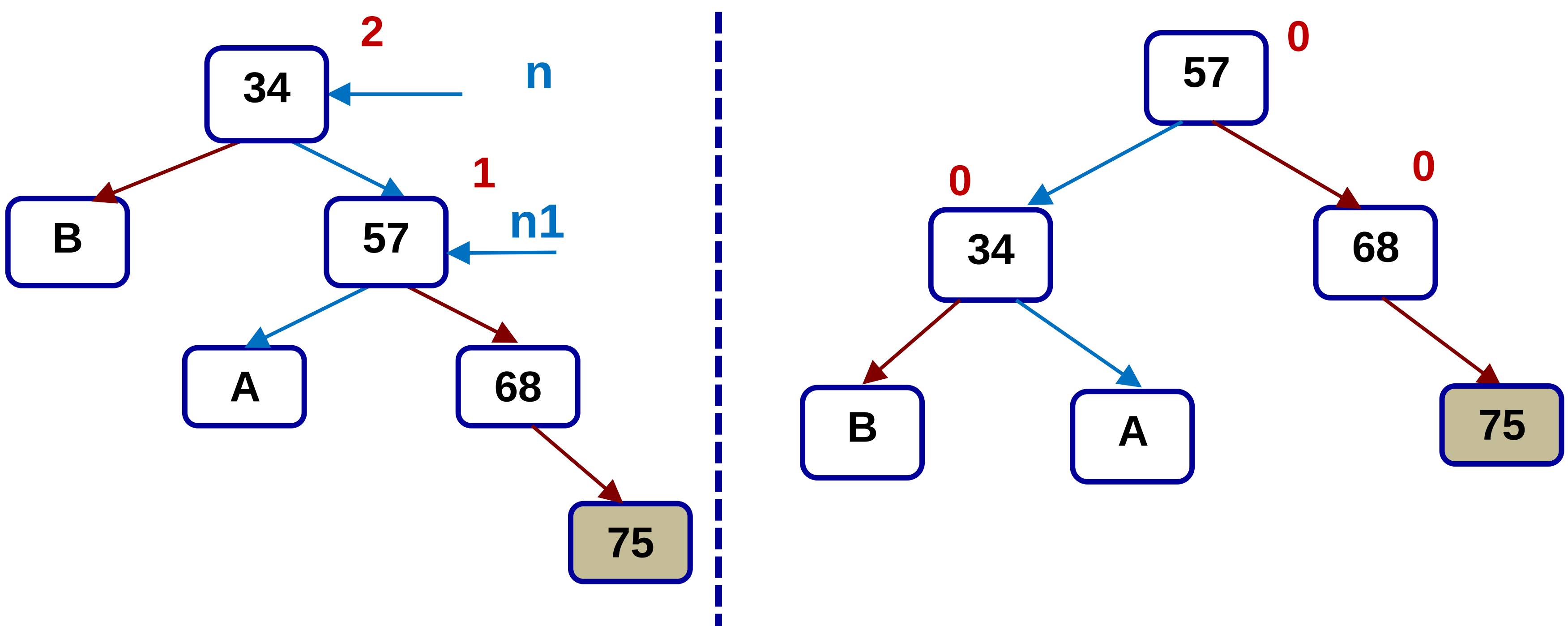
Se realiza cuando el **factor de equilibrio de un nodo es 2**, es decir, su subárbol derecho es dos unidades más alto que el izquierdo, y, además, **la raíz del subárbol derecho tiene un FE de 1**, es decir, el **árbol está cargado a la derecha**.

ÁRBOLES AVL – Rotaciones Simples

Rotación simple Derecha-Derecha:

1. Pasar subárbol izquierdo del nodo **n1** como subárbol derecho del nodo **n**.
2. El árbol **n** pasa a ser el subárbol izquierdo del nodo **n1**.
3. El nodo **n1** toma la posición del nodo **n** (la entrada al árbol es ahora el nodo **n1**).

```
n->hdch = n1->hizq;
n1->hizq = n
n = n1
```



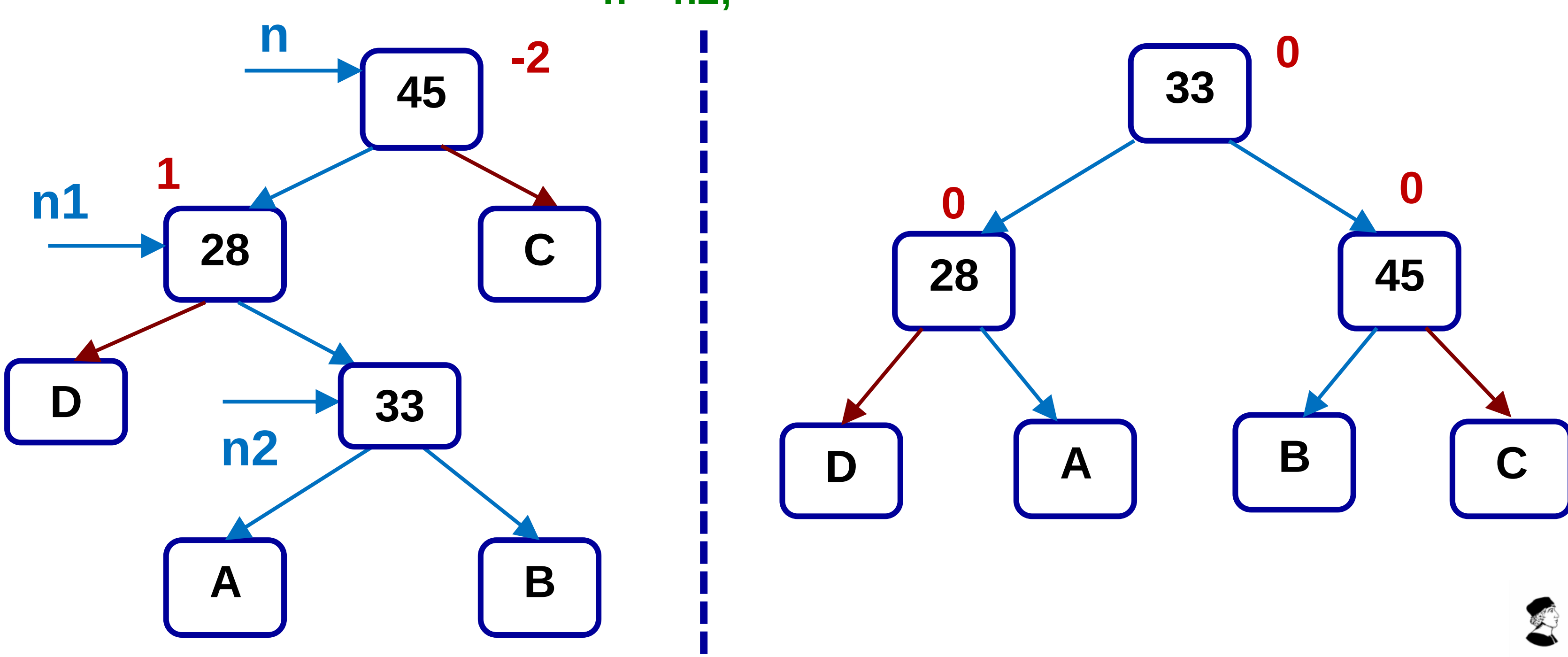
Rotación doble Izquierda-Derecha (ID):

Se usa cuando el **subárbol izquierdo sea dos unidades más alto que el derecho**, es decir, cuando su **FE sea -2**, y, además, la **raíz del subárbol izquierdo** tenga **un FE de 1**, es decir, esté **cargado a la derecha**.

Rotación doble a la Izquierda-Derecha (ID):

1. Hacemos una rotación simple de **n1** a la izquierda.
2. Hacemos una rotación simple de **n** a la derecha.

```
n1->hdch = n2->hizq;
n2->hizq = n1;
n->hizq = n2->hdch;
n2->hdch = n;
n = n2;
```



Rotación doble Derecha-Izquierda (DI):

Se usa cuando el **subárbol derecho sea dos unidades más alto que el izquierdo**, es decir, cuando su **FE sea 2**, y, además, la **raíz del subárbol derecho** tenga **un FE de -1**, es decir, esté **cargado a la izquierda**.

Rotación doble Derecha-Izquierda (DI):

1. Hacemos una rotación simple de **n1** a la derecha.
2. Hacemos una rotación simple de **n** a la izquierda.

```
n1->hizq = n2->hdch;
n2->hdch = n1;
n->hdch = n2->hizq;
n2->hizq = n;
n = n2;
```