

ESTRUCTURAS DE DATOS

TIPOS ABSTRACTOS DE DATOS ARBORESCENTES

EL TAD **Árbol Binario**

Manuel Montenegro Montes
Departamento de Sistemas Informáticos y Computación
Facultad de Informática – Universidad Complutense de Madrid

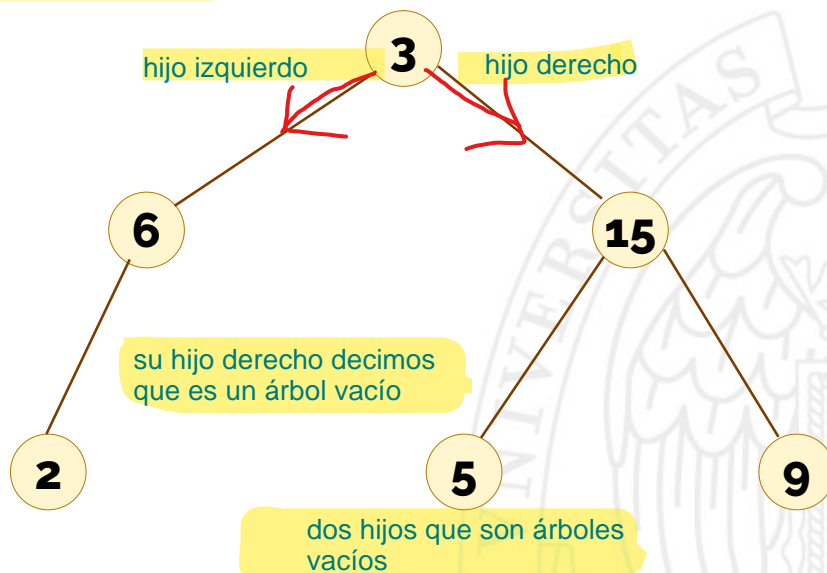
Árboles binarios

- Un árbol binario es un árbol de aridad 2.
- Cada nodo tiene **2** hijos, algunos de los cuales pueden ser vacíos.

Máximo grado es 2

Máximo grado de nuestro árbol es 2

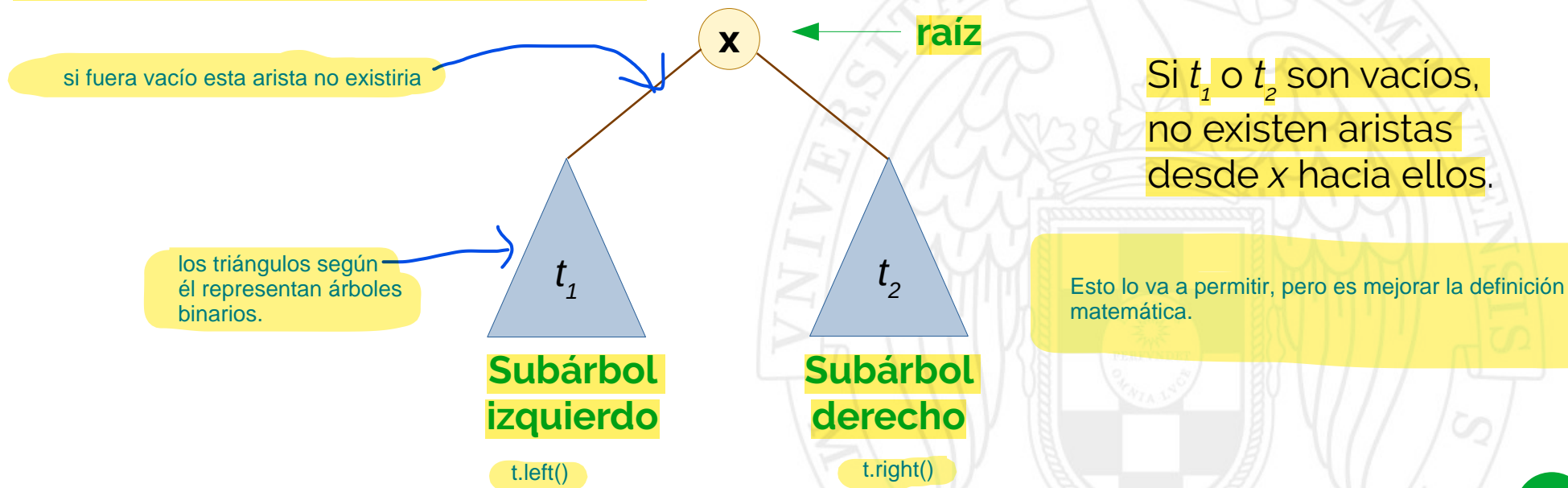
Vamos, que cada nodo admite 2 hijos



Definición inductiva de un árbol binario

- Caso base:** Un grafo sin nodos es un **árbol vacío**. `tree.empty() == true;`

- Caso recursivo:** Si t_1 y t_2 son árboles binarios, y x es un elemento, entonces lo siguiente es un árbol binario:



Definición inductiva de un árbol binario

- Un árbol binario T es un conjunto finito tal que:
 - $T = \emptyset$, o bien
 - $T = \{x\} \uplus T_1 \uplus T_2$, donde T_1 y T_2 son árboles.

x es la raíz,

T_1 es el subárbol izquierdo, y

T_2 es el subárbol derecho.

Definición matemática. No es necesario sabérsela.

Operaciones en el TAD Árbol Binario

- Constructoras:

- Crear un árbol vacío: **create_empty**.
- Crear una hoja: **create_leaf**. es redundante, podría obtenerse a partir de CREATE_TREE PERO PASANDO 2 ÁRBOLES VACÍOS.
- Crear un árbol a partir de una raíz y dos hijos: **create_tree**.

- Observadoras:

- Determinar si el árbol es vacío: **empty**.
- Obtener la raíz si el árbol no es vacío: **root**.
- Obtener el hijo izquierdo, si existe: **left**.
- Obtener el hijo derecho, si existe: **right**.

Para los árboles binarios funcionan muy bien la recursión, en concreto el divide y vencerás.

COMO PODEMOS VER, EN ESTE TAD NO HAY OPERACIONES MUTADORAS.

Operaciones constructoras

$\{ true \}$

`create_empty()` $\rightarrow (T: \text{ArBin})$

$\{ T = \boxed{\quad} \}$

Vacío.

$\{ true \}$

`create_leaf`($x: \text{Elem}$) $\rightarrow (T: \text{ArBin})$

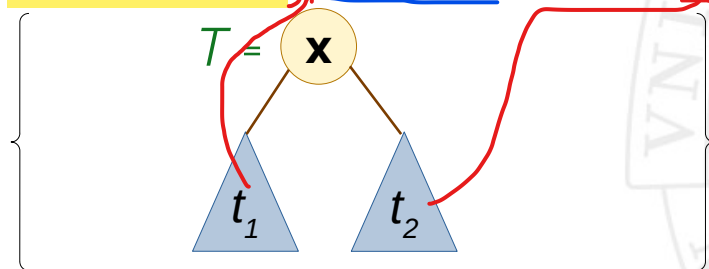
$\{ T = \boxed{x} \}$

nodo sin hijos, una hoja.

$\left\{ T_1 = \triangle t_1 \quad T_2 = \triangle t_2 \right\}$

Elemento raíz.

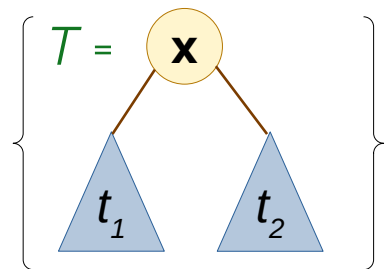
`create_tree`($T_1: \text{ArBin}, x: \text{Elem}, T_2: \text{ArBin}$) $\rightarrow (T: \text{ArBin})$



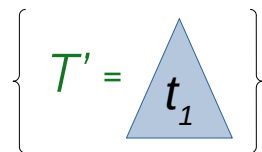
recibe 2 árboles y un elemento

x es el elemento del que cuelgan los otros dos árboles.

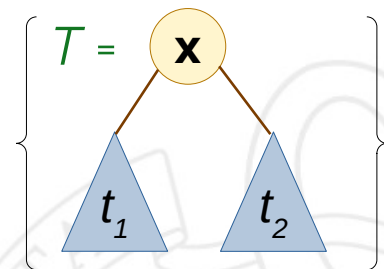
Operaciones observadoras



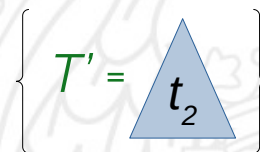
left($T: \text{ArBin}$) $\rightarrow (T': \text{ArBin})$



Devuelve el hijo izquierdo

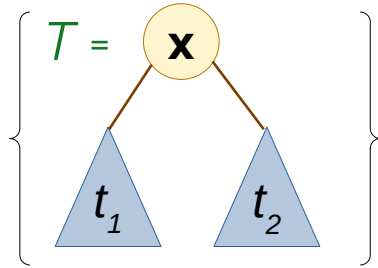


right($T: \text{ArBin}$) $\rightarrow (T': \text{ArBin})$



Devuelve el hijo derecho.

Operaciones observadoras



root($T: \text{ArBin}$) $\rightarrow (e: \text{elem})$

$\{ e = x \}$

$\{ \text{true} \}$

empty($T: \text{ArBin}$) $\rightarrow (b: \text{bool})$

$\{ b \Leftrightarrow T = \text{—} \}$

Es decir, que el árbol sea vacío.

LA ECUACIÓN QUE DETERMINA EL NÚMERO DE NODOS DE UN ÁRBOL BINARIO ES:

$2^{\text{nivel} - 1}$