

ESTRUCTURAS DE DATOS

TIPOS ABSTRACTOS DE DATOS ARBORESCENTES

El TAD Conjunto

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

Conjuntos

- Un **conjunto** es una colección de elementos del mismo tipo.
- ¿Cuál es la diferencia entre un conjunto y una lista?

Lista

- El orden de los elementos es relevante:

$$[1, 4, 5] \neq [4, 5, 1]$$

- Pueden contener elementos duplicados:

$$[1, 4, 4, 5] \neq [1, 4, 5]$$

Conjunto

- No existe el concepto de orden entre elementos:

$$\{1, 4, 5\} = \{4, 5, 1\}$$

- La existencia de duplicados es irrelevante:

$$\{1, 4, 5\} \cup \{4\} = \{1, 4, 5\}$$

Modelo de conjuntos

- Varias formas de implementar un conjunto.
- Cuando el conjunto está implementado y tan solo tenemos que utilizarlo, pensamos en términos del **modelo**.
- Cada instancia del TAD Conjunto representa un **conjunto finito**.

$$\{x_1, x_2, x_3, \dots, x_n\}$$

Operaciones en el TAD conjunto

- Constructoras:
 - Crear un conjunto vacío: ***create_empty***
- Mutadoras:
 - Añadir un elemento al conjunto: ***insert***
 - Eliminar un elemento del conjunto: ***erase***
- Observadoras:
 - Averiguar si un elemento está en el conjunto: ***contains***
 - Saber si el conjunto está vacío: ***empty***
 - Saber el tamaño del conjunto: ***size***

Operaciones constructoras y mutadoras

$\{ \text{true} \}$

create_empty() $\rightarrow (S: \text{Set})$

$\{ S = \emptyset \}$

$\{ \text{true} \}$

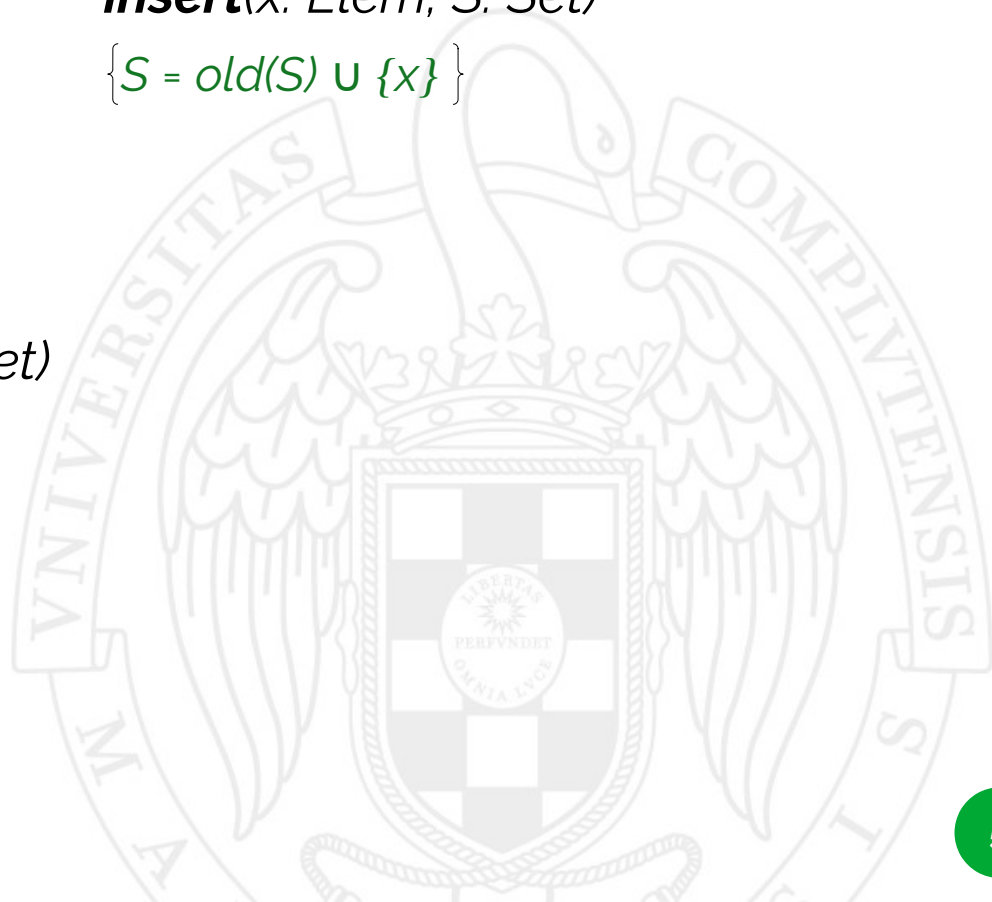
insert($x: \text{Elem}, S: \text{Set}$)

$\{ S = \text{old}(S) \cup \{x\} \}$

$\{ \text{true} \}$

erase($x: \text{Elem}, S: \text{Set}$)

$\{ S = \text{old}(S) - \{x\} \}$



Operaciones observadoras

$\{ \text{true} \}$

contains($x: \text{Elem}, S: \text{Set}$) $\rightarrow (b: \text{bool})$

$\{ b \Leftrightarrow x \in S \}$

$\{ \text{true} \}$

empty($S: \text{Set}$) $\rightarrow (b: \text{bool})$

$\{ b \Leftrightarrow S = \emptyset \}$

$\{ \text{true} \}$

size($S: \text{Set}$) $\rightarrow (n: \text{int})$

$\{ n = |S| \}$



Interfaz en C++

```
class set {  
public:  
    set();  
    set(const set &other);  
    ~set();  
  
    void insert(const T &elem);  
    void erase(const T &elem);  
  
    bool contains(const T &elem) const;  
    int size() const;  
    bool empty() const;  
  
private:  
    // ???  
};
```



Dos implementaciones

- Mediante **listas**.
- Mediante **árboles binarios de búsqueda**.

