Iniciación a la programación con Java

Estructuras encadenadas

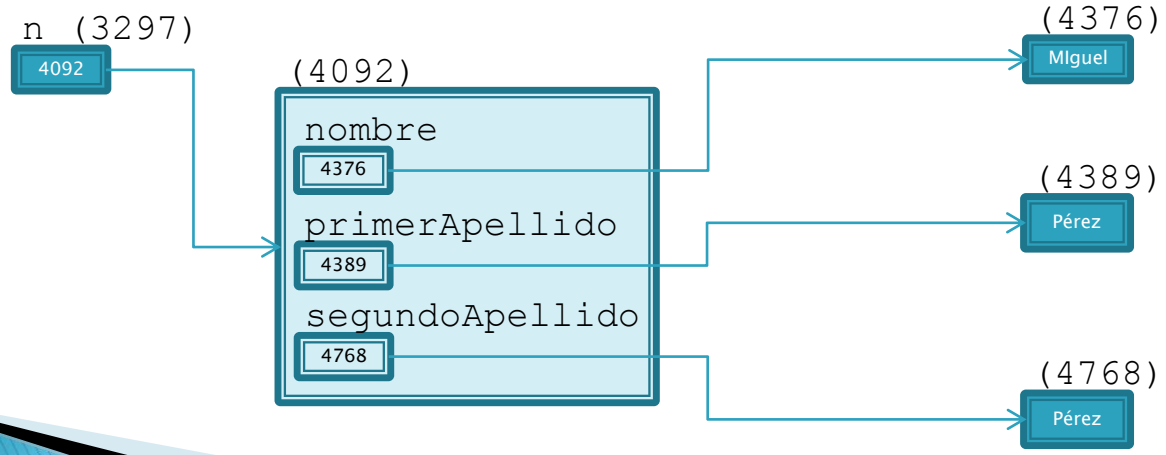
tilde.dis.ulpgc.es
Tecnología e Innovación para Labrar
el Desarrollo Educativo

Estructuras encadenadas

- Referencias a objetos

Referencias

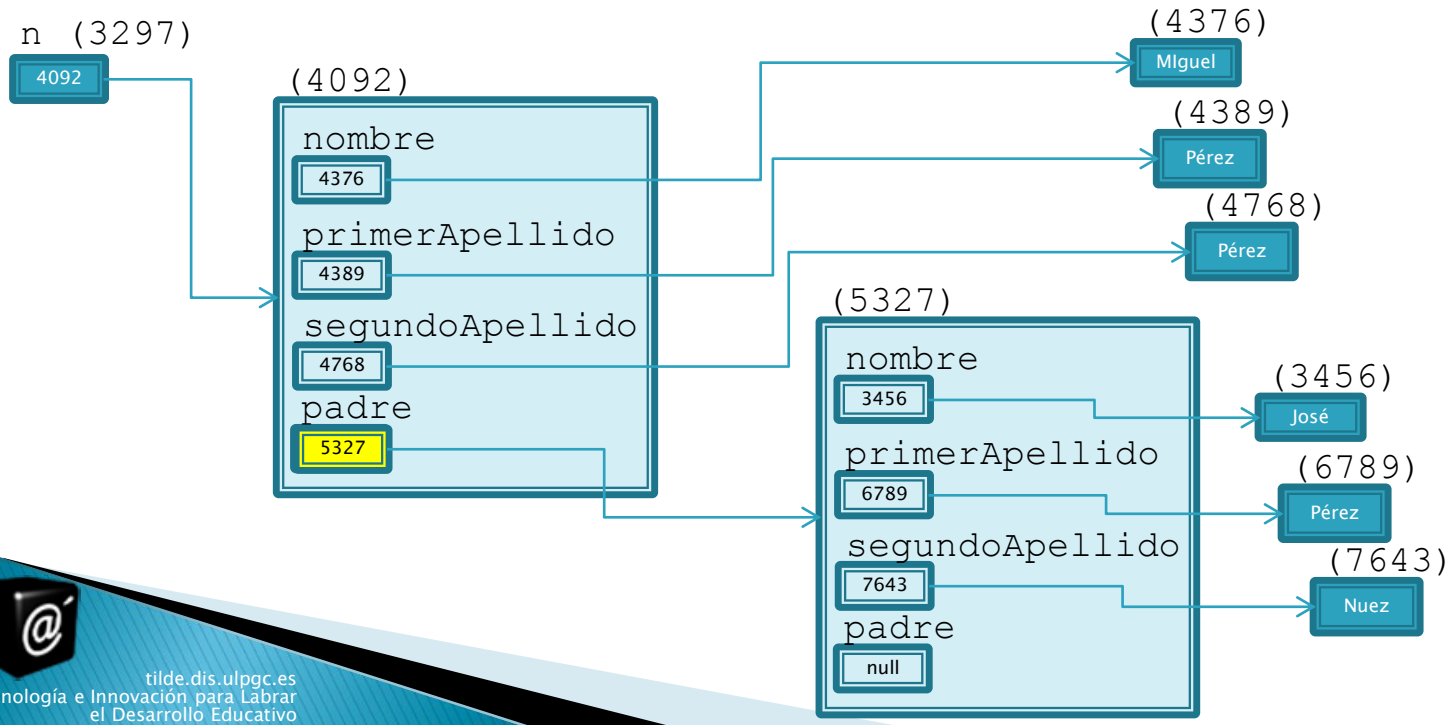
```
public class Persona{
    private String nombre;
    private String primerApellido;
    private String segundoApellido;
    …
    public Persona(String n, String a1, String a2)…
    …
}

Persona n = new Persona("Miguel", "Pérez", "Pérez");
```



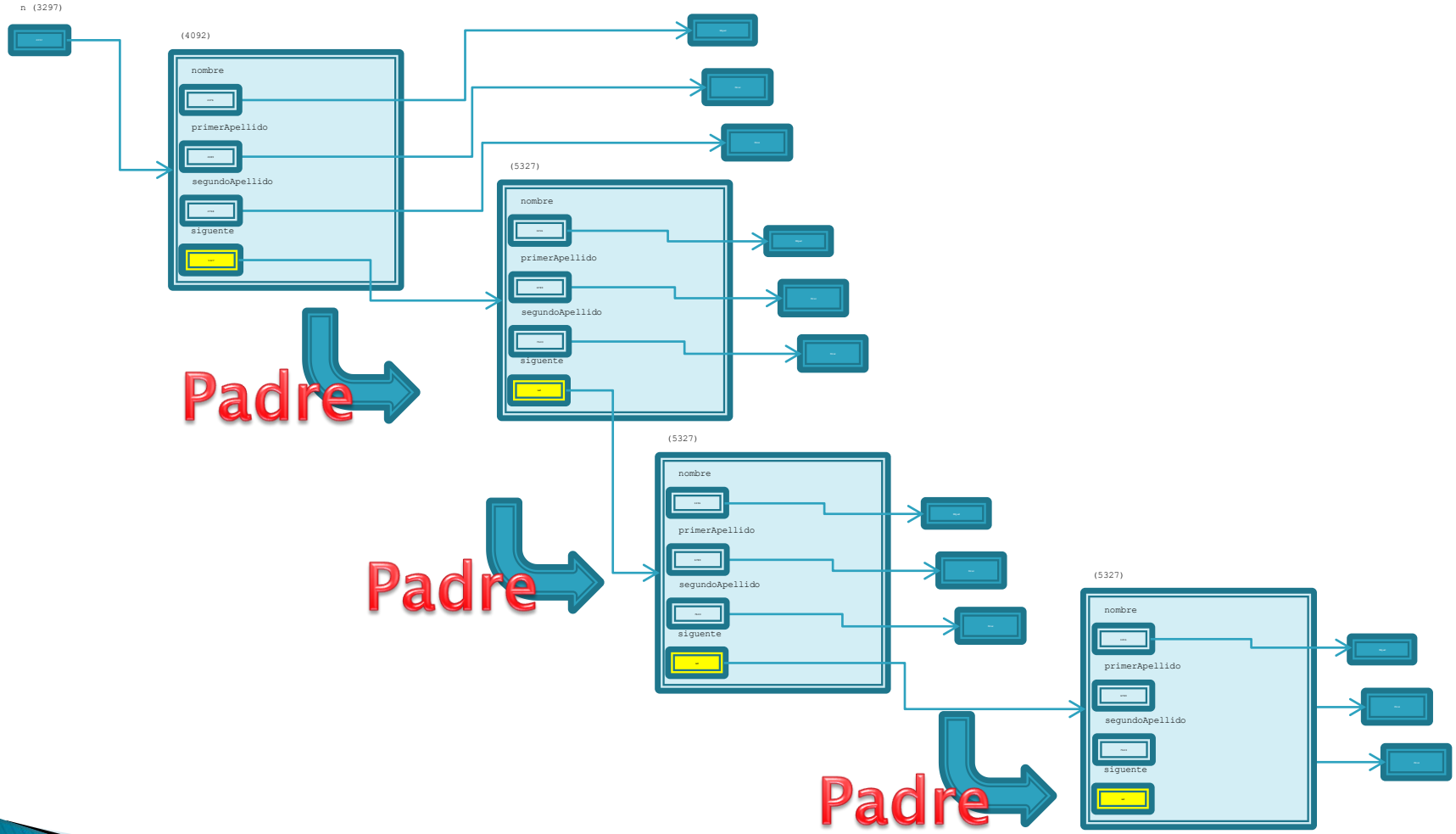
Referencias recursivas

```
public class Persona {
    …
    private Persona padre;
    …
    public void setPadre(Persona p) {padre = p;}
    public Persona getPadre() {return padre;} …
}
```

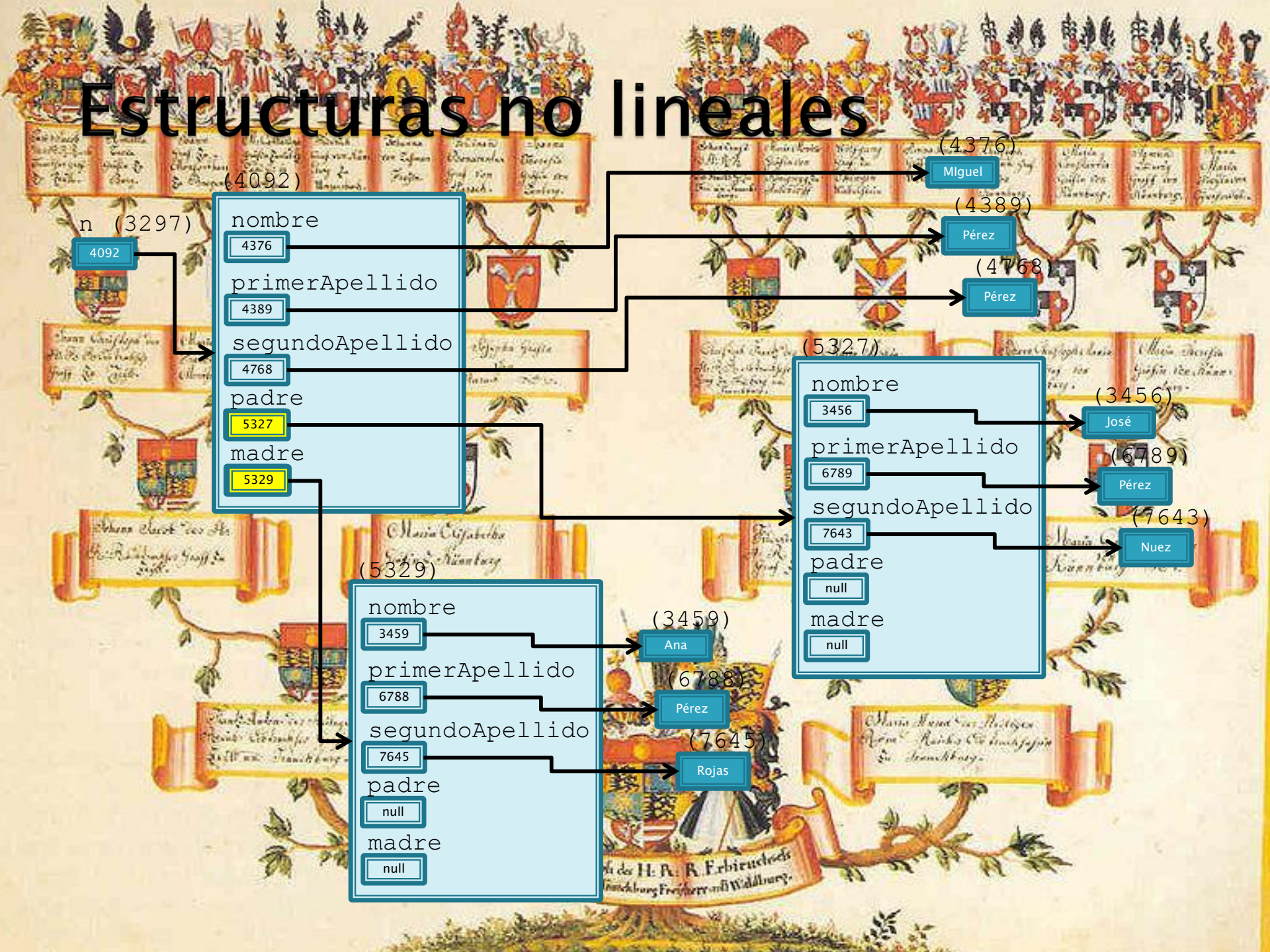
```
Persona n = new Nombre("Miguel", "Pérez", "Pérez");
n.setPadre(new Persona("José", "Pérez", "Nuez"));
```



Estructuras lineales



Estructuras no lineales



Iniciación a la programación con Java

Estructuras encadenadas

tilde.dis.ulpgc.es
Tecnología e Innovación para Labrar
el Desarrollo Educativo

Aplicaciones de las estructuras encadenadas

- Aparte de para aplicaciones específicas, las estructuras encadenadas pueden ser muy útiles para la implementación de contenedores (listas, tablas, conjuntos, pilas, colas,…)

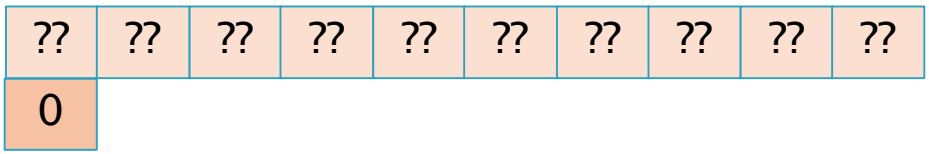
Ejemplo: lista de enteros

- Implementación

Usando un array

```
public class ArrayListOfInt {
    private int[] data;
    private int size = 0;

    public ArrayListOfInt(int capacity) {
        data = new int[capacity];
    }
    …
}

ArrayListOfInt l = new ArrayListOfInt(10);
```

Hace falta definir la capacidad a priori

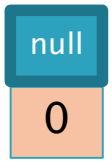


Usando una estructura encadenada

```
public class LinkedListOfInt {
    private class Node {
        int value;
        Node next;
    }
    Node data;
    int size = 0;
    …
}

LinkedListOfInt l = new LinkedListOfInt();
```

No hace falta conocer la capacidad a priori

Cada valor tiene asociado espacio adicional para los enlaces



Iniciación a la programación con Java

Estructuras encadenadas

tilde.dis.ulpgc.es
Tecnología e Innovación para Labrar
el Desarrollo Educativo

Ejemplo: lista de enteros

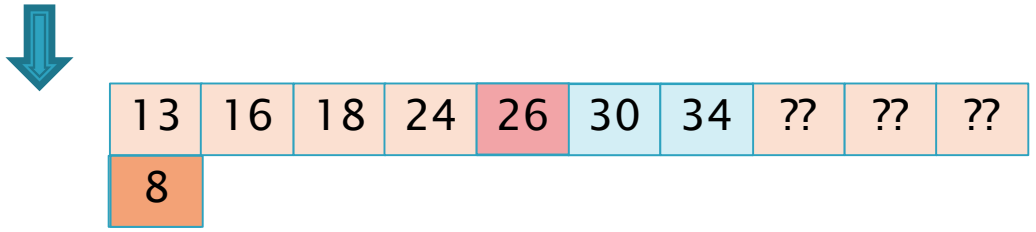
- Inserción

Inserción en un array

```
public void insert(int element) {
    int i;
    for (i = 0; i < size && data[i] < element; i++);
    System.arraycopy(data, i, data, i+1, size - i);
    data[i] = element;
    size++;
}
```

```
l.insert(26);
```

Hay que desplazar elementos

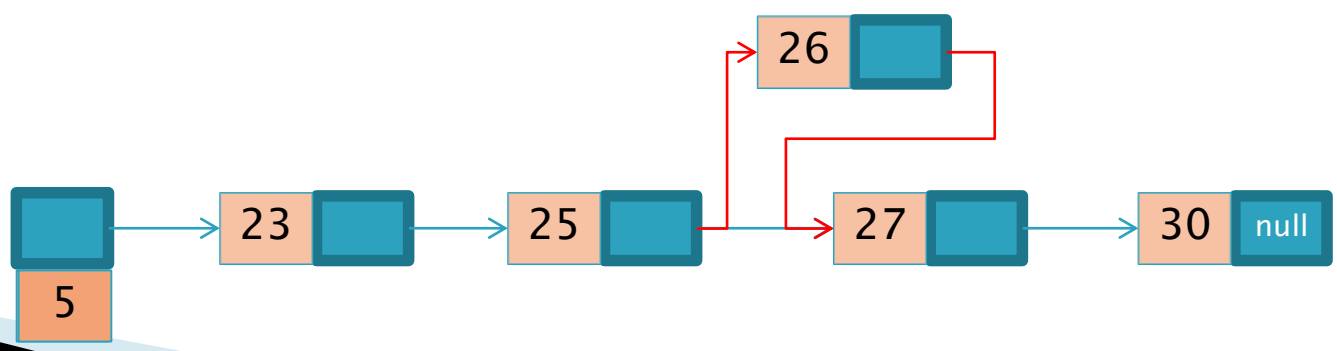


Inserción en un lista encadenada

```
public void insert(int element) {
    Node prev = null;
    Node current = data;
    while (current != null && current.value < element) {
        prev = current;
        current = current.next;
    }
    Node newNode = new Node();
    newNode.value = element;
    newNode.next = current;
    if(prev == null) {
        data = newNode;
    } else {
        prev.next = newNode;
    }
    size++;
}
```

No hace falta desplazar elementos

```
l.insert(26);
```



Iniciación a la programación con Java

Estructuras encadenadas

tilde.dis.ulpgc.es
Tecnología e Innovación para Labrar
el Desarrollo Educativo

Ejemplo: lista de enteros

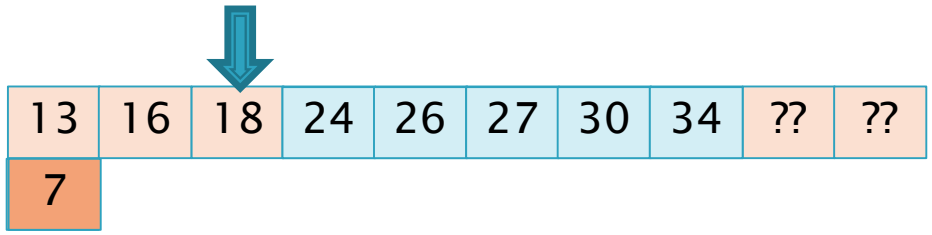
- Extracción

Extracción en un array

```
public void deleteAt(int pos) {
    System.arraycopy(data, pos+1, data, pos, size - pos - 1);
    size--;
}
```

```
l.deleteAt(2);
```

Hay que desplazar elementos



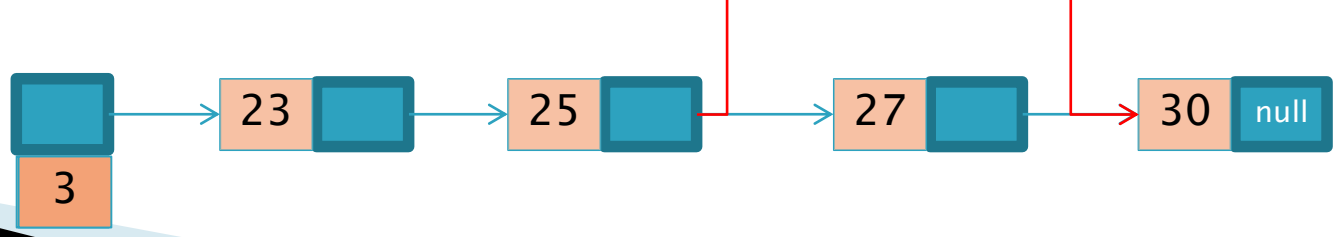
Extracción en un lista encadenada

```
public void deleteAt(int pos) {
    Node prev = null;
    Node current = data;
    int auxPos = 0;
    while (auxPos < pos) {
        prev = current;
        current = current.next;
        auxPos++;
    }
    if (pos == 0) {
        data = data.next;
    }
    else {
        prev.next = current.next;
    }
    size--;
}
```

No hace falta desplazar elementos

No hay acceso directo a la posición

```
l.deleteAt(2);
```



Iniciación a la programación con Java

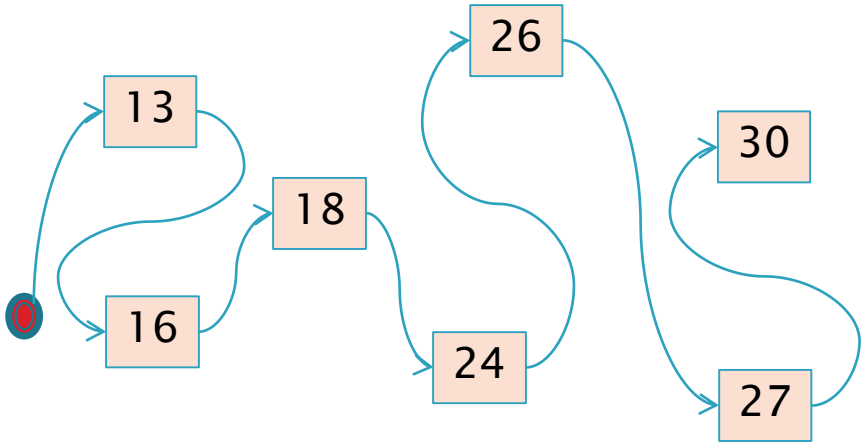
Estructuras encadenadas

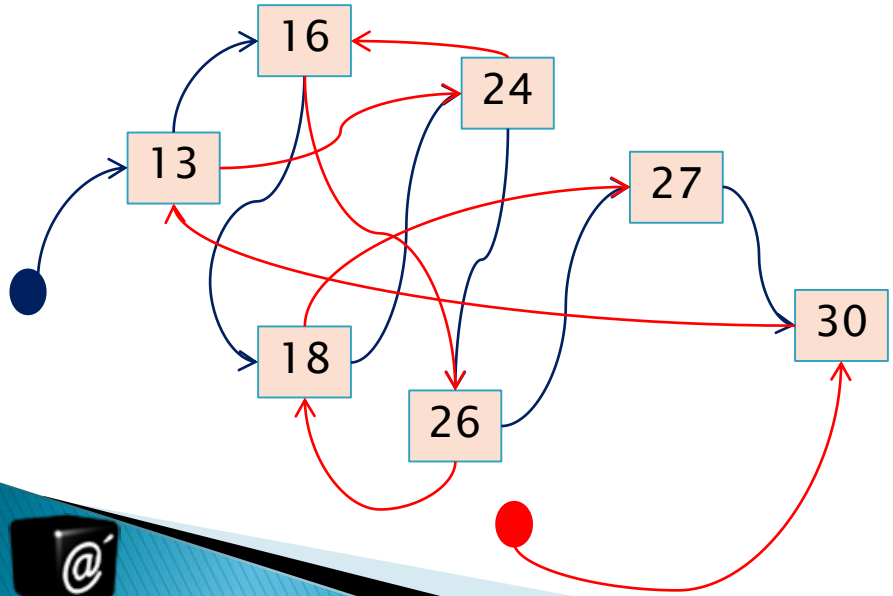
tilde.dis.ulpgc.es
Tecnología e Innovación para Labrar
el Desarrollo Educativo

Estructuras encadenadas

- Resumen de ventajas e inconvenientes

Ventajas





- Las estructuras encadenadas independizan las relaciones entre los elementos frente a su posición física lo cual permite:
  - Insertar y extraer elementos sin tener que cambiar de sitio a los demás
  - Establecer relaciones múltiples y complejas entre elementos

Inconvenientes

- Al necesitarse espacio para los enlaces además de para la información, parece que se consume más memoria, pero:
  - Esto no siempre es cierto, depende de lo que necesitásemos si tuviésemos que reservar la capacidad máxima.
  - Su importancia frente al volumen de información puede ser relativa.
- Frente a un array, se pierde el acceso directo por posición
  - El cual no siempre es necesario, por lo que la importancia de su pérdida depende del problema concreto.

Iniciación a la programación con Java

Estructuras encadenadas

tilde.dis.ulpgc.es
Tecnología e Innovación para Labrar
el Desarrollo Educativo