



# Conceptos de Algoritmos Datos y Programas

# CADP – TIPOS DE DATOS - LISTA



Creación de una lista.

Agregar nodos al comienzo de la lista.

Recorrido de una lista.

Agregar nodos al final de la lista.

Buscar un elemento en una lista

Eliminar un elemento de una lista

Insertar un elemento en una lista ordenada



# CADP – TEMAS



- Operación de BUSCAR un ELEMENTO



**Significa recorrer la lista desde el primer nodo buscando un valor que puede o no estar. Se debe tener en cuenta si la lista está o no ordenada.**

### LISTA Desordenada

- Se debe recorrer toda la lista (en el peor de los casos), y detener la búsqueda en el momento que se encuentra el dato buscado o en el que la lista se terminó.

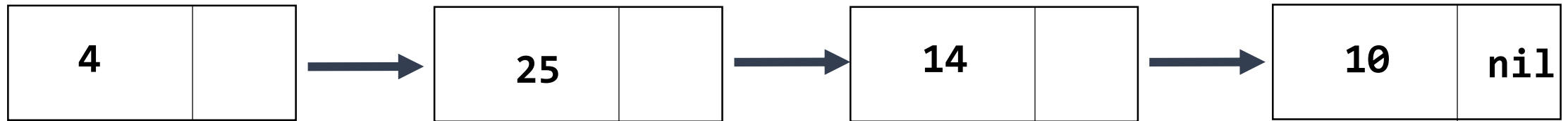
### LISTA Ordenada

- Se debe recorrer la lista teniendo en cuenta el orden. La búsqueda se detiene cuando se termina la lista o el elemento buscado es mayor al elemento actual.



## BUSQUEDA LISTA DESORDENADA

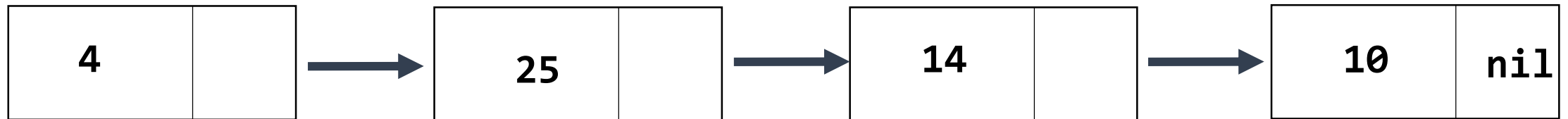
aux



Pri

num = 14

aux



Pri

num = 3



## BUSQUEDA LISTA DESORDENADA

Comienzo a recorrer la lista desde el nodo inicial.

mientras ((no sea el final de la lista)y(no encuentre el elemento))

    si (es el elemento buscado) entonces

        detengo la búsqueda

    sino

        avanzo al siguiente elemento

*Qué módulo  
utilizo?*



```
Program uno;
```

```
Type listaE= ^datosEnteros;
```

```
    datosEnteros= record  
        elem:integer;  
        sig:listaE;  
    end;
```

```
Var
```

```
    pri: listaE;  
    num:integer;
```

```
Begin
```

```
    crear (pri);  
    cargar (pri); //se dispone  
    read (num);  
    if (buscar(pri,num)) then write ("el elemento existe");
```

```
End.
```



```
function buscar (pI: listaE; valor:integer):boolean;  
Var  
  aux:listaE;  
  encontré:boolean;  
  
Begin  
  encontré:= false;  
  aux:= pI;  
  while ((aux <> nil) and (encontré = false)) do  
    begin  
      if (aux^.elem = valor) then  
        encontré:=true  
      else  
        aux:= aux^.sig;  
      end;  
    buscar:= encontré;  
  end;  
end;
```

*Funciona si la  
lista que recibo  
es vacía?*

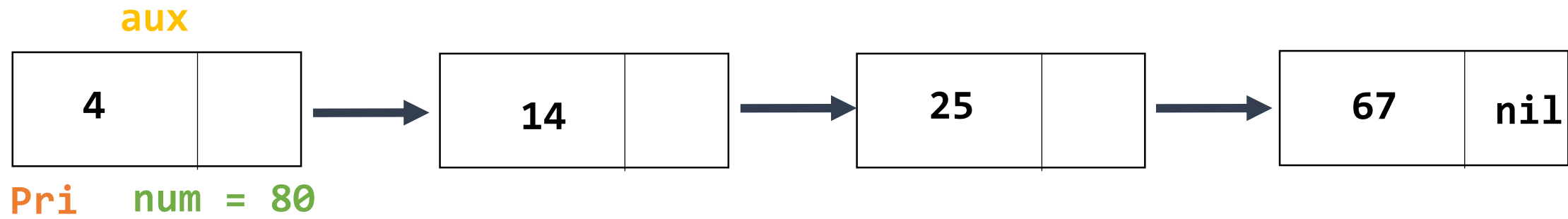
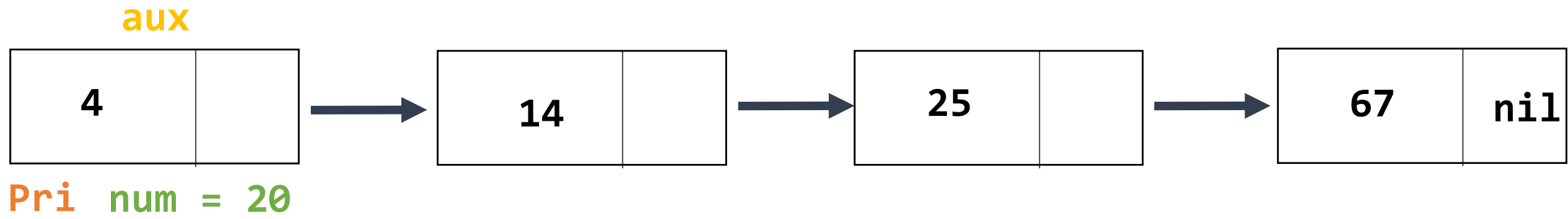
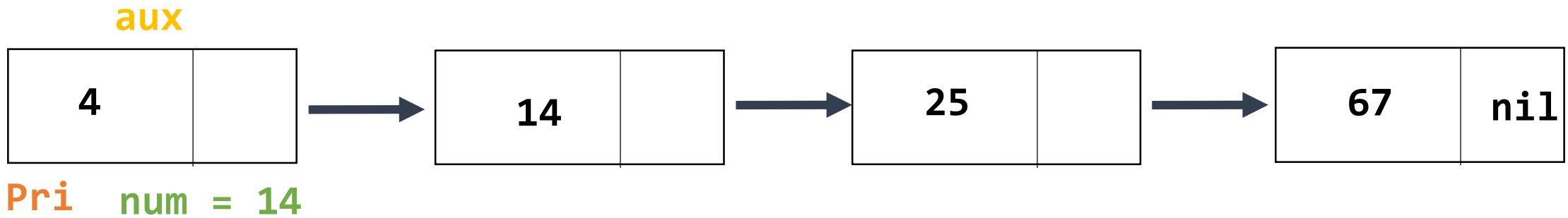
*Necesito  
usar aux?*

**Qué modifiko si la lista está  
ordenada?**





## BUSQUEDA LISTA ORDENADA





```
function buscar (pI: listaE; valor:integer):boolean;
```

```
Var
```

```
    aux:listaE;
```

```
    encontré:boolean;
```

```
Begin
```

```
    encontré:= false;
```

```
    aux:= pI;
```

```
    while ((aux <> nil) and (aux^.elem < valor)) do
```

```
        begin
```

```
            aux:= aux^.sig;
```

```
        end;
```

```
        if (aux <> nil) and (aux^.elem = valor) then encontré:= true;
```

```
        buscar:= encontré;
```

```
end;
```

*Funciona si la lista que  
recibo es vacía?*

*Necesito usar  
aux?*

*Es necesario respetar el  
orden de las condiciones?*

*Necesito el chequeo  
del final?*

**Buscar en una lista tiene las mismas  
características que buscar en un vector**