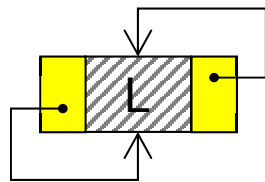
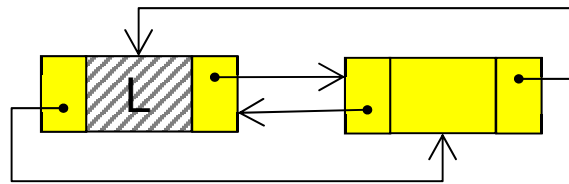
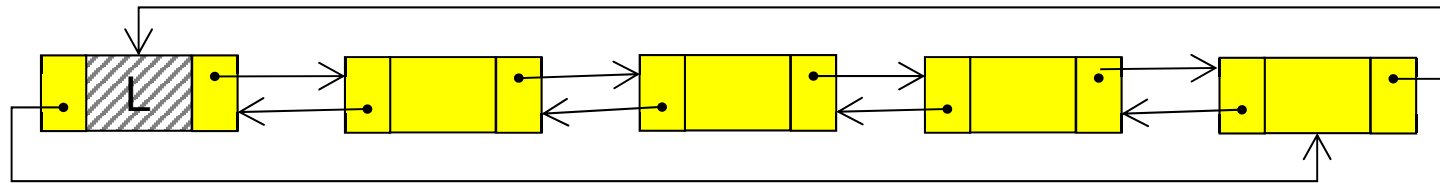
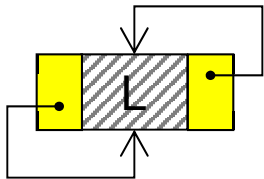
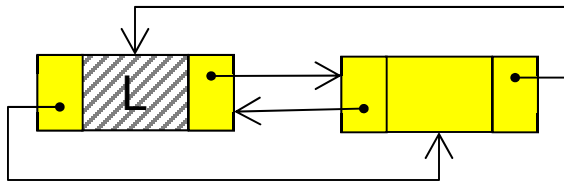
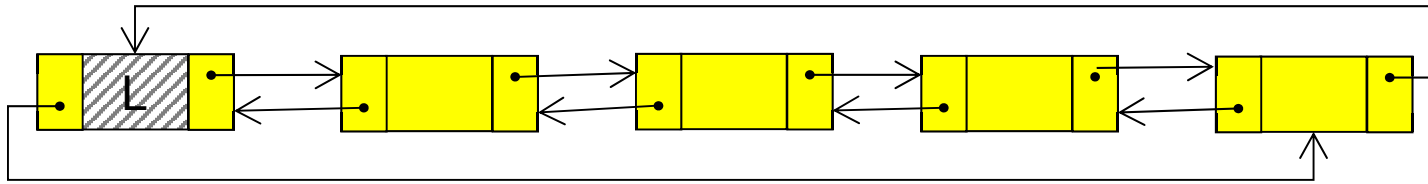


Lista Doblemente Ligada Circular Con Encabezado Dummy (LDLCCED)



```
typedef _____ tipo_dato;  
  
typedef struct tipo_nodo {  
    tipo_dato elem;  
    struct tipo_nodo *ant;  
    struct tipo_nodo *sig;  
    } tipo_nodo;  
  
typedef tipo_nodo tipo_lista;  
  
typedef tipo_nodo *tipo_pos;
```



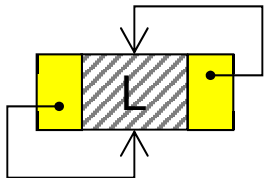
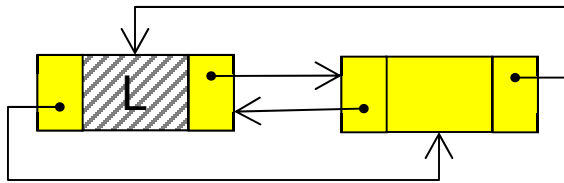
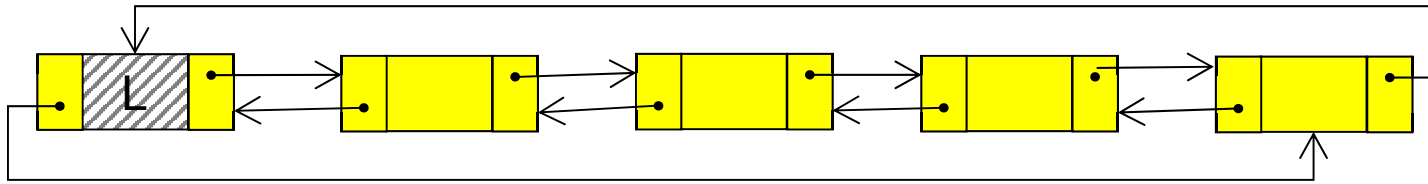
función: **inicializa**

recibe: *lista*

regresa: nada

$lista.ant = lista$

$lista.sig = lista$



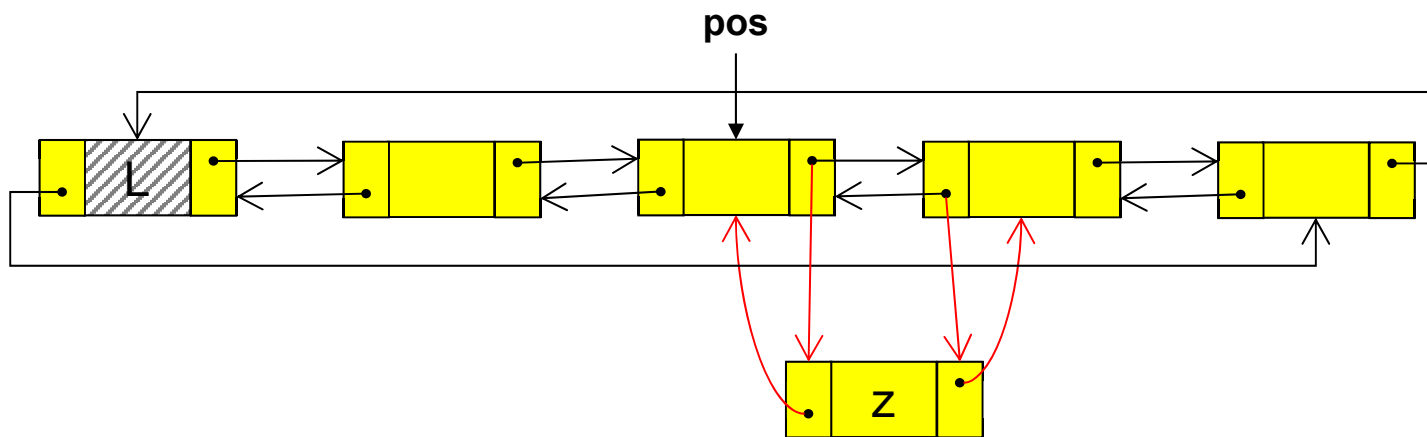
función: **vacía**
 recibe: *lista*
 regresa: booleano

$\neg lista.sig = lista?$

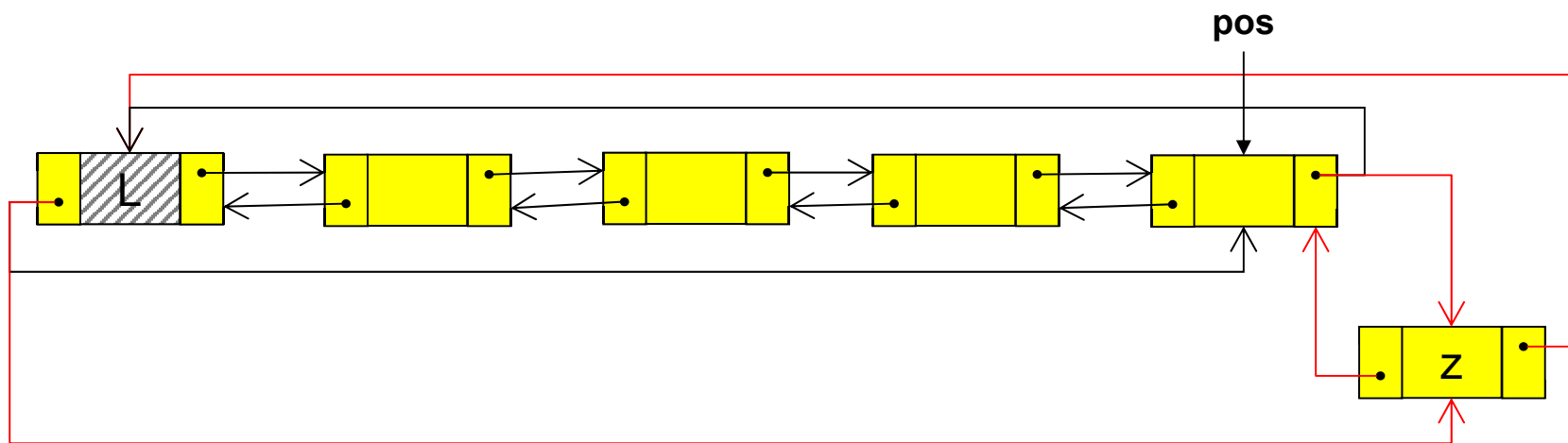
Sí: regresar: *verdadero*

No: regresar: *falso*

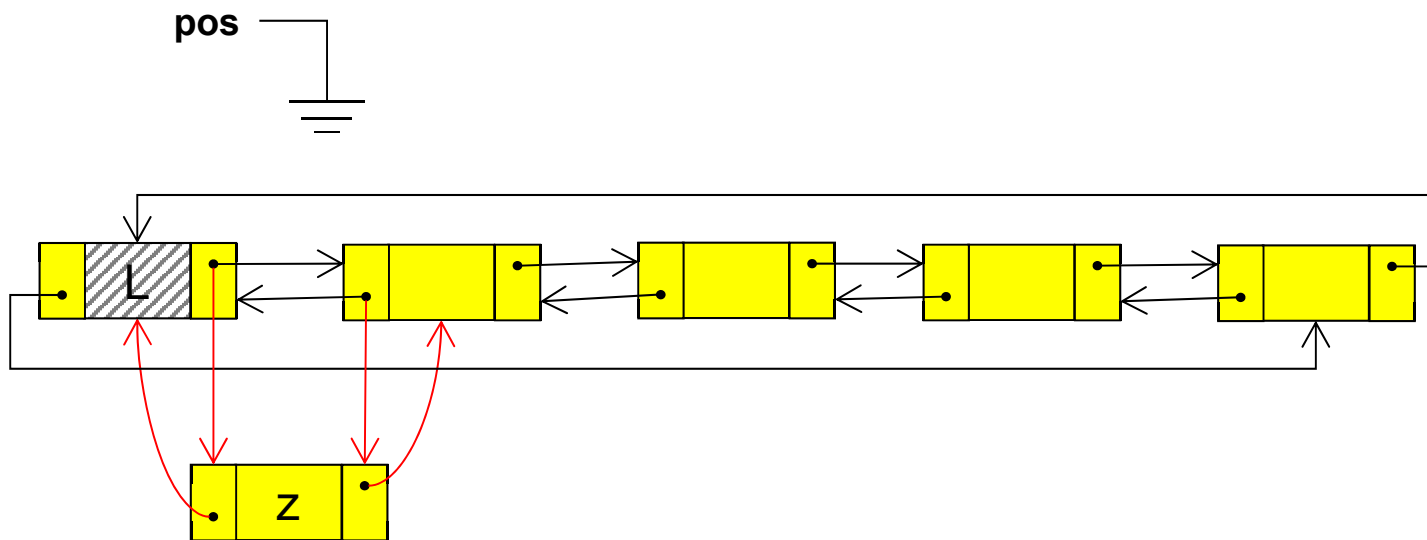
Inserción
en cualquier posición



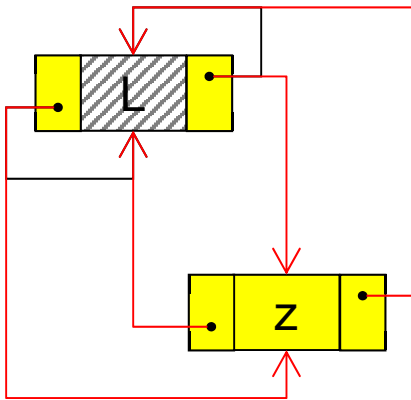
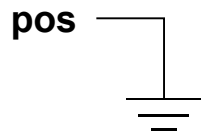
Inserción en cualquier posición

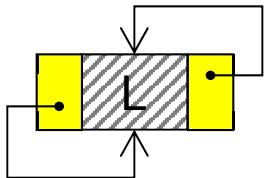
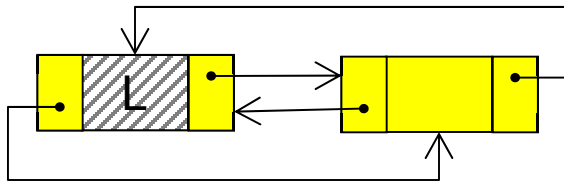
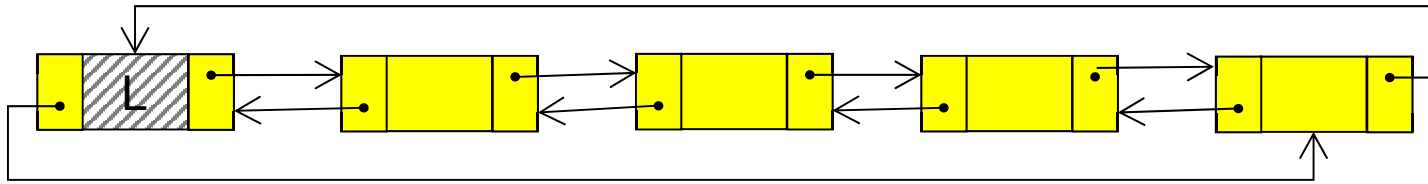


Insertión
en cualquier posición



Insertión
en cualquier posición





función: **inserta**
 recibe: *elem*, *pos*, *lista*
 regresa: nada

aux = nuevo nodo
 $aux \rightarrow elem = elem$

$\neg pos = \text{NULO?}$

pos = *lista*

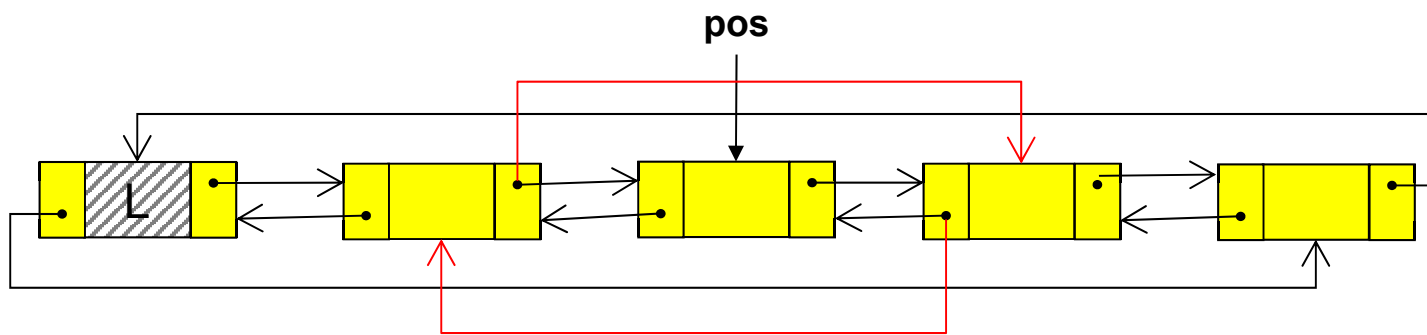
$aux \rightarrow ant = pos$

$aux \rightarrow sig = pos \rightarrow sig$

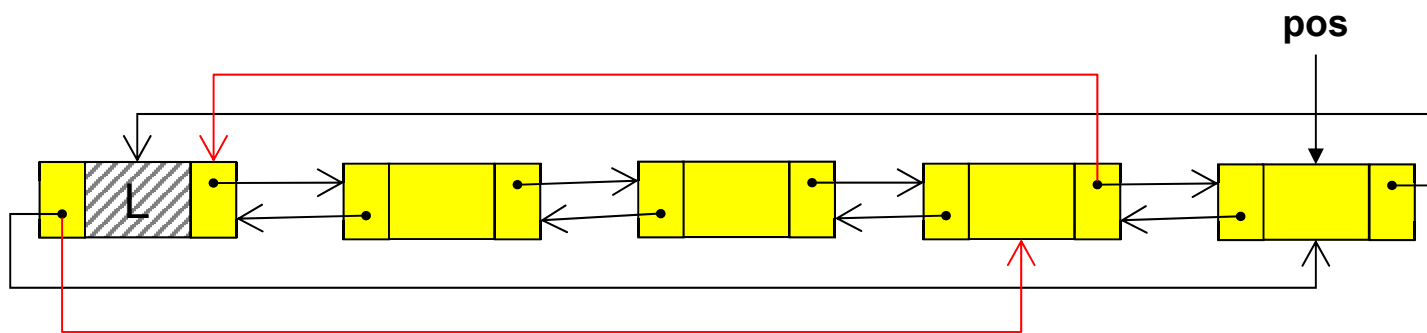
$pos \rightarrow sig \rightarrow ant = aux$

$pos \rightarrow sig = aux$

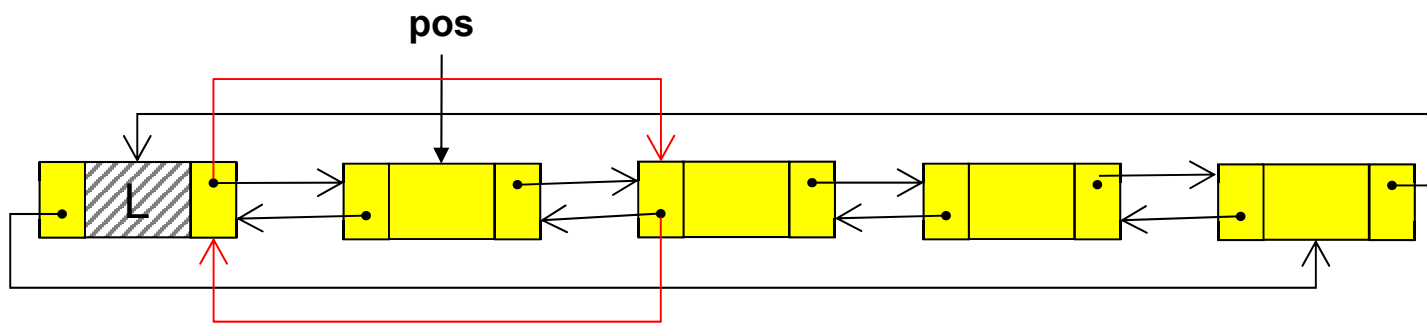
Eliminación en cualquier posición



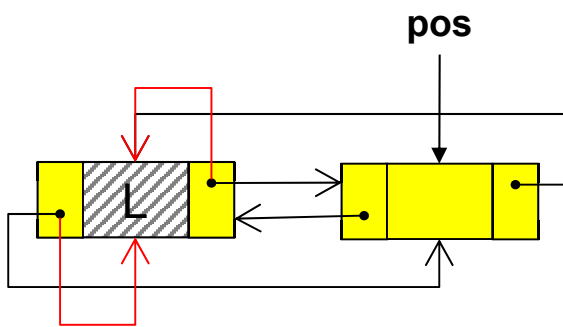
Eliminación
en cualquier posición

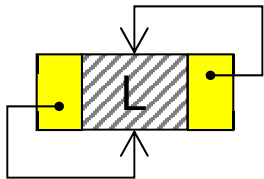
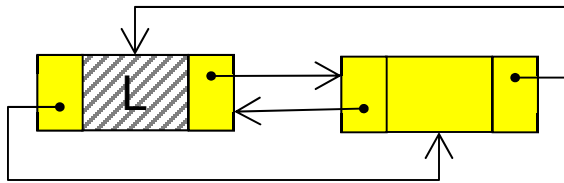
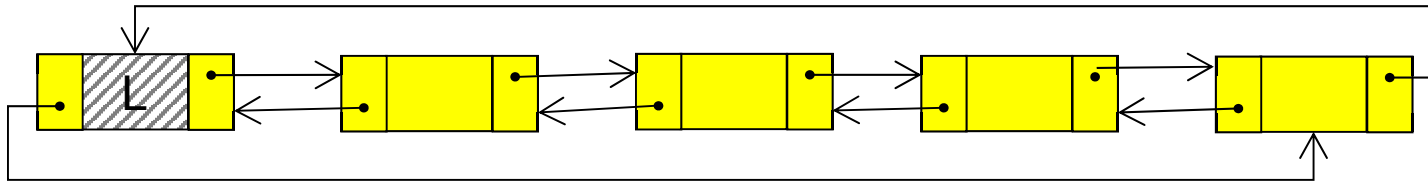


Eliminación en cualquier posición



Eliminación
en cualquier posición





función: **elimina**
 recibe: *pos*, *lista*
 regresa: nada

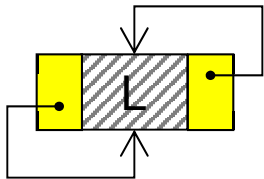
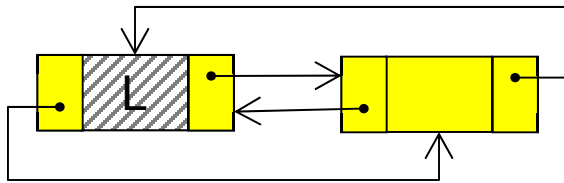
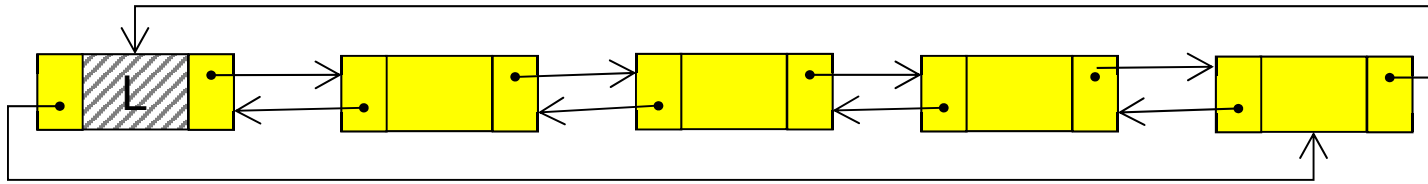
¿**vacía**(*lista*) ó *pos* = NULO?

Sí: terminar

$pos \rightarrow ant \rightarrow sig = pos \rightarrow sig$

$pos \rightarrow sig \rightarrow ant = pos \rightarrow ant$

liberar espacio de memoria de *pos*



función: ***primero***

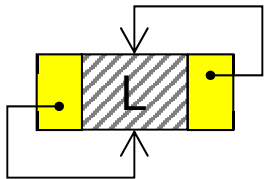
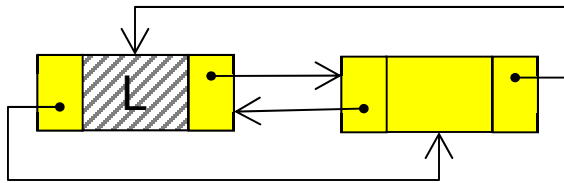
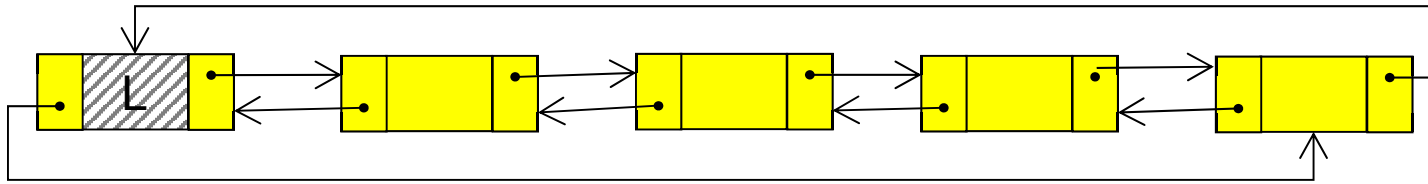
recibe: *lista*

regresa: posición

¿vacía(*lista*)?

Sí: regresar: NULO
terminar

regresar: *lista.sig*

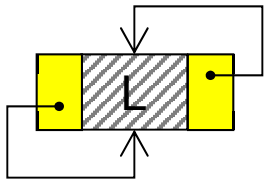
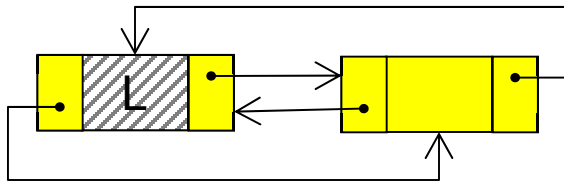
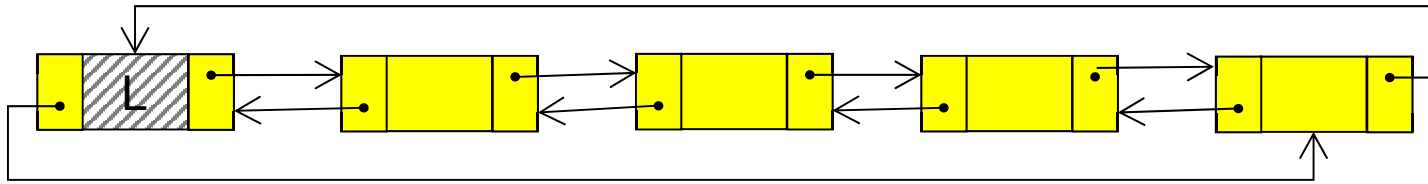


función: **último**
 recibe: *lista*
 regresa: posición

¿vacía(*lista*)?

Sí: regresar: NULO
 terminar

regresar: *lista* → ant



función. **anterior**

recibe: pos , $lista$

regresa: posición

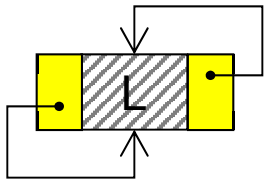
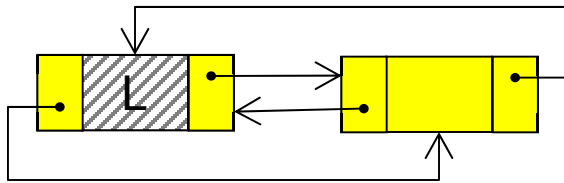
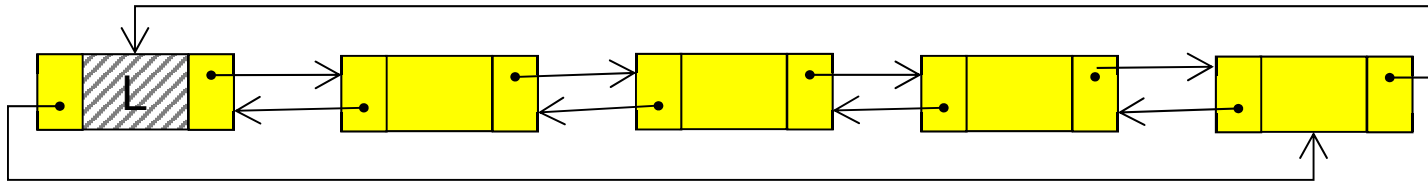
¿**vacía**($lista$) ó $pos = \text{NULO}$?

Sí: regresar: NULO
terminar

¿ $pos \rightarrow ant = lista$?

Sí: regresar: $lista.ant$

No: regresar: $pos \rightarrow ant$



función. **anterior**

recibe: pos , $lista$

regresa: posición

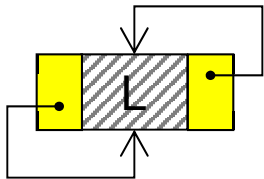
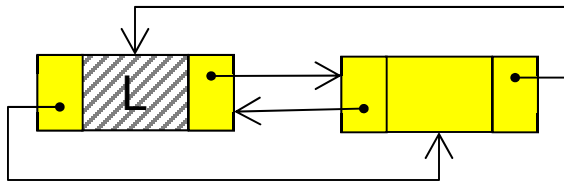
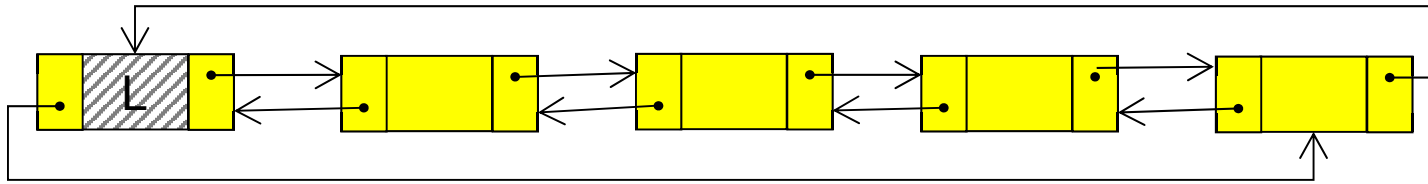
¿vacía($lista$) ó $pos = \text{NULO}$?

Sí: regresar: NULO
terminar

¿ $pos \rightarrow \text{sig} = lista$?

Sí: regresar: $lista.\text{sig}$

No: regresar: $pos \rightarrow \text{sig}$



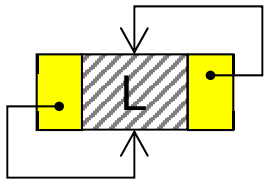
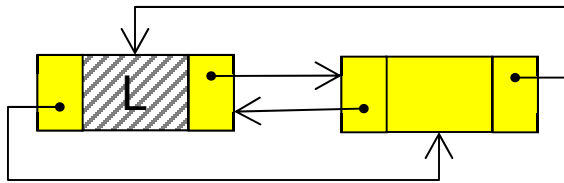
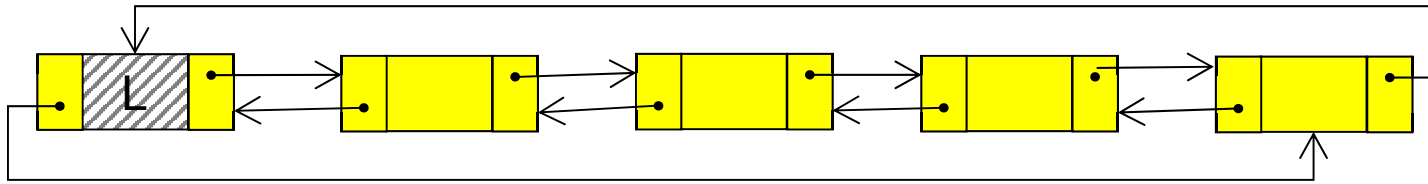
función: **localiza**
 recibe: *elem*, *lista*
 regresa: posición

```

aux = lista.sig
mientras aux ≠ lista
  ¿aux→elem = elem?
    Sí: regresar: aux
    salir
  aux = aux→sig

```

regresar: NULO



función: **recupera**

recibe *pos*, *lista*

regresa: elemento

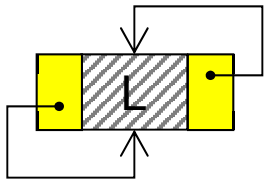
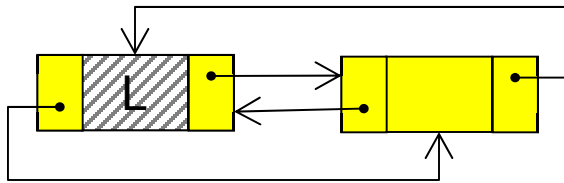
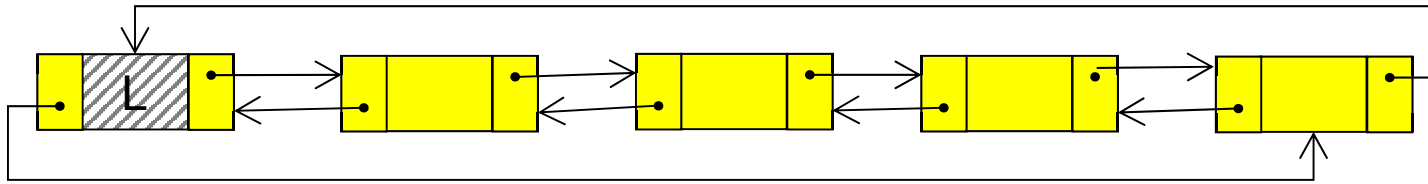
¿vacía(lista) ó *pos* = NULO?

Sí: ¡error de excepción!

Insuficiencia de datos
terminar



No: regresar: *pos* → elem



función: **imprime**

recibe: *lista*

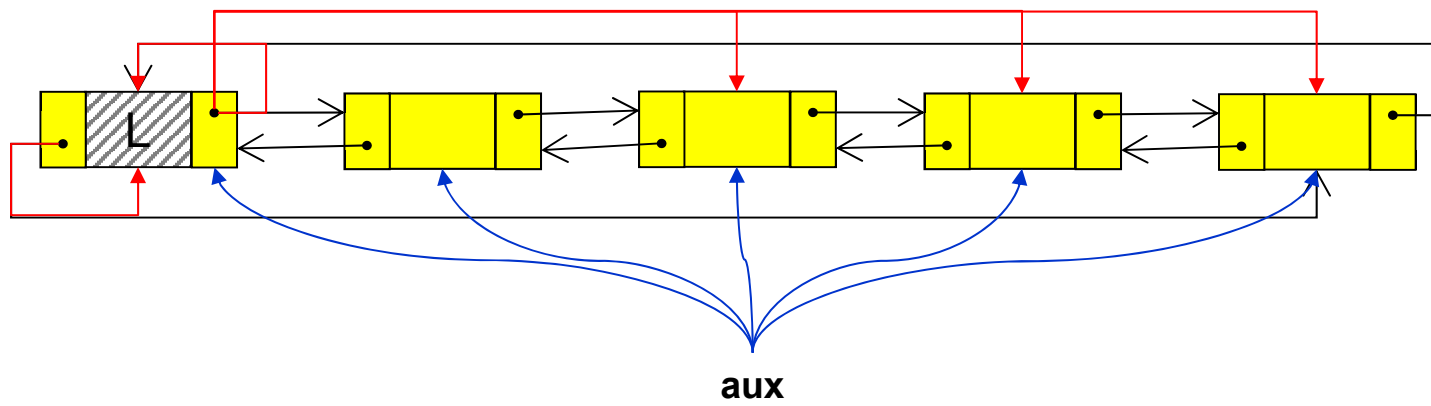
regresa: nada

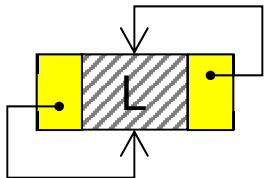
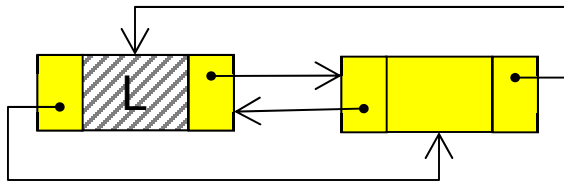
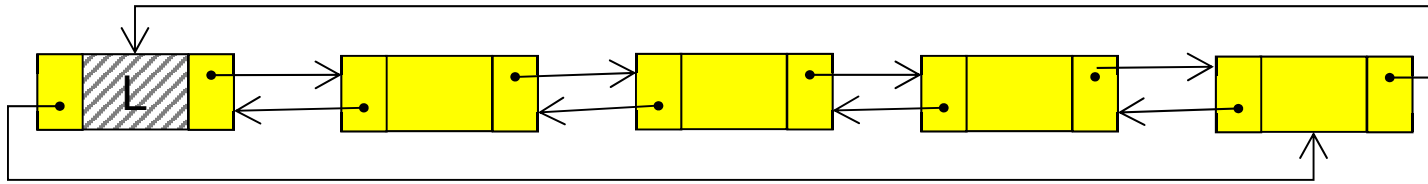
$aux = lista.sig$

mientras $aux \neq lista$

imprimir $aux \rightarrow elem$

$aux = aux \rightarrow sig$





función: **anula**

recibe: *lista*

regresa: nada

mientras $lista.sig \neq lista$

$aux = lista.sig$

$lista.sig = aux \rightarrow sig$

liberar espacio de memoria de aux

$lista.ant = lista$