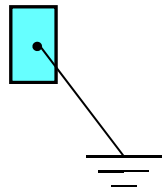
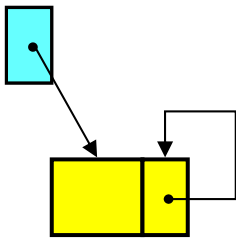
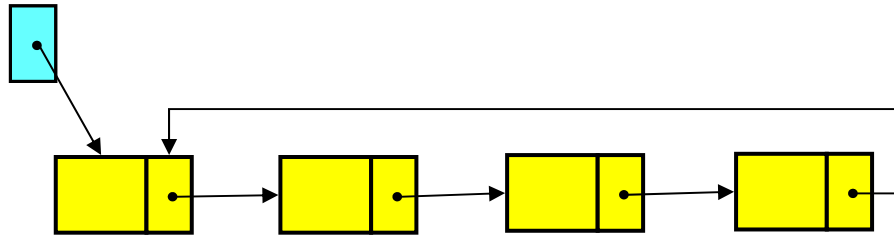
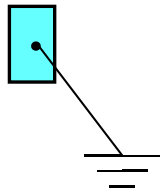
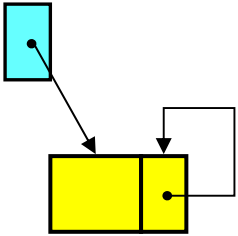
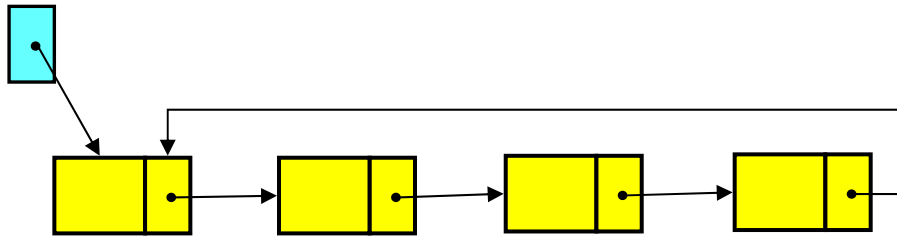


Lista Simplemente Ligada Circular Sin Encabezado (LSLCSE)



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

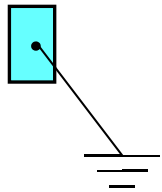
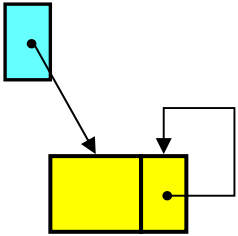
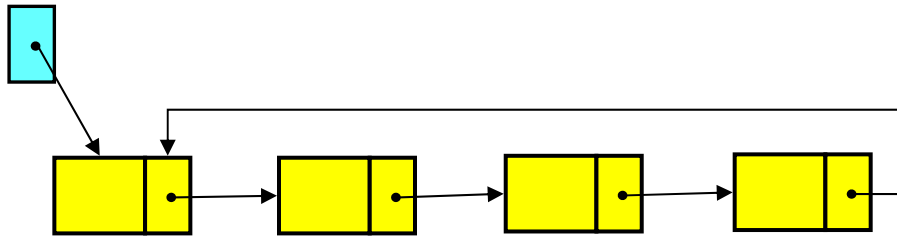


función: **inicializa**

recibe: *lista*

regresa: nada

lista = NULO



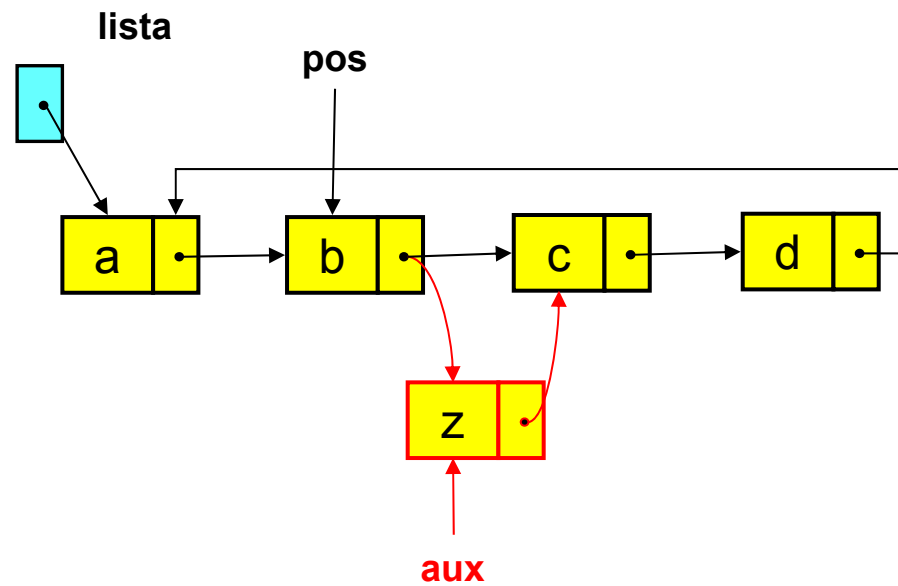
función: **vacía**
 recibe: *lista*
 regresa: booleano

¿lista = NULO?

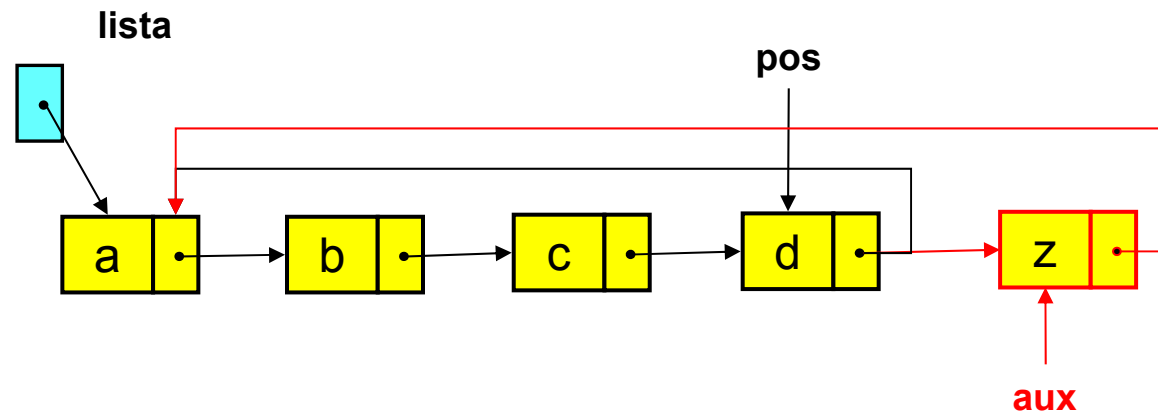
Sí: regresar: *verdadero*

No: regresar: *falso*

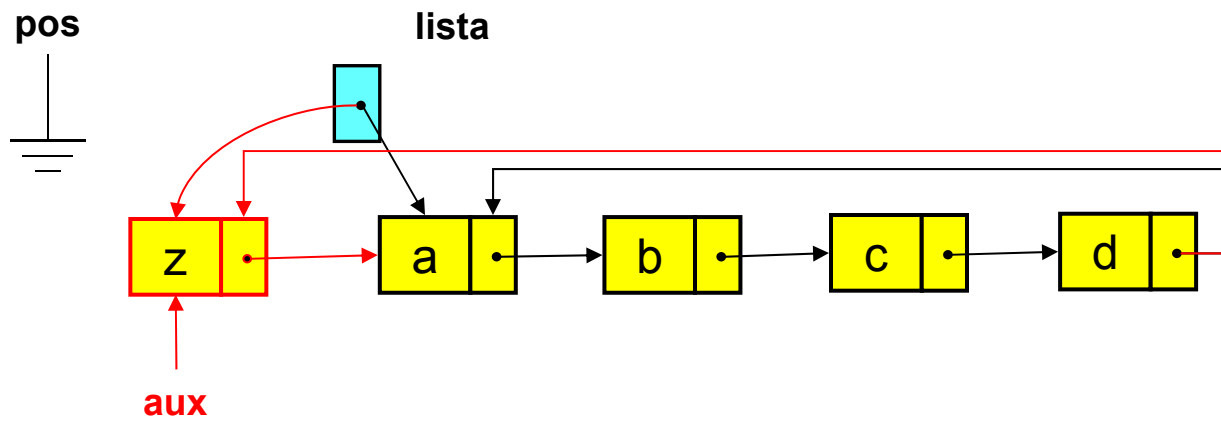
Insertión
en cualquier posición,
excepto al principio



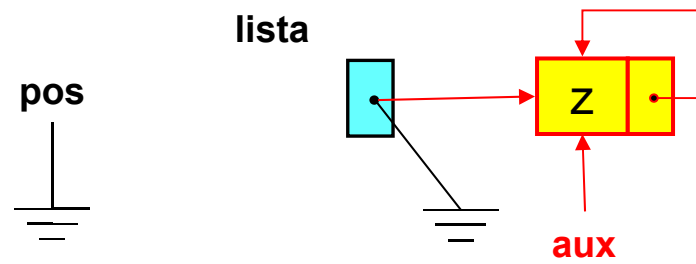
Insertión
en cualquier posición,
excepto al principio

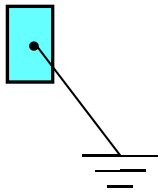
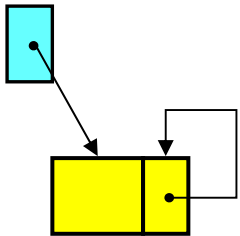
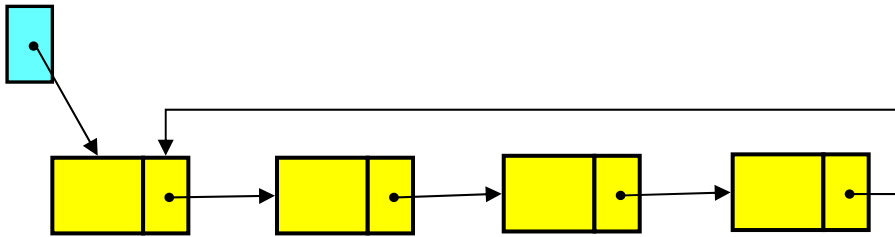


Inserción al principio



Inserción al principio





función: **inserta**

recibe: *elem*, *pos*, *lista*

regresa: nada

$aux = \text{nuevo nodo}$

$aux \rightarrow \text{elem} = elem$

¿**vacía**(*lista*)?

Sí: $aux \rightarrow \text{sig} = aux$

$lista = aux$

No: ¿*pos* = NULO?

Sí: $aux \rightarrow \text{sig} = lista$

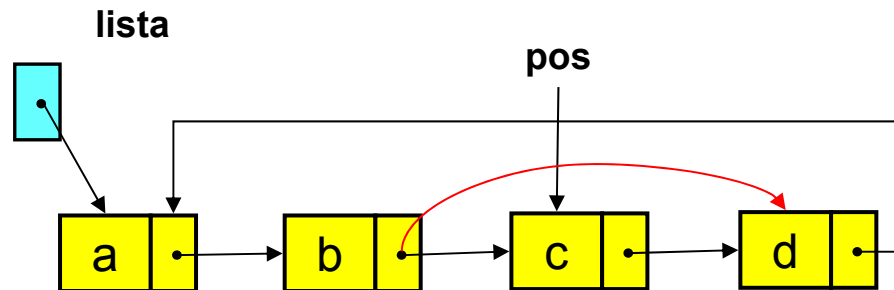
ultimo(*lista*) $\rightarrow \text{sig} = aux$

$lista = aux$

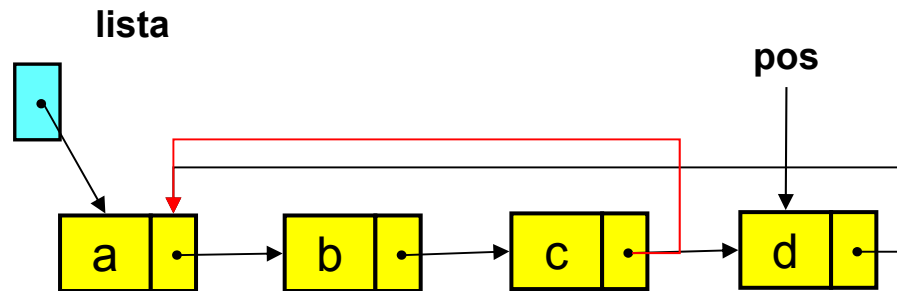
No: $aux \rightarrow \text{sig} = pos \rightarrow \text{sig}$

$pos \rightarrow \text{sig} = aux$

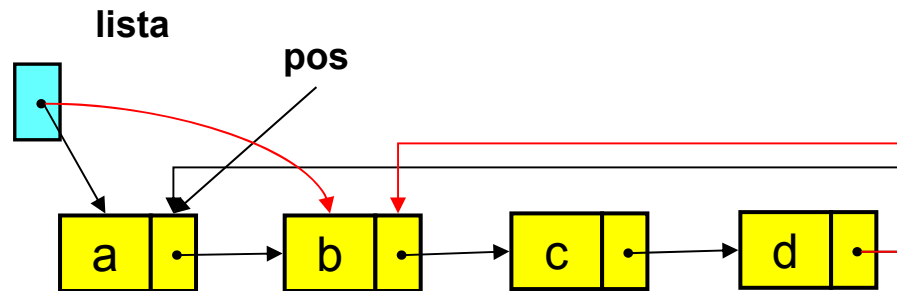
Eliminación
en cualquier posición,
excepto al principio



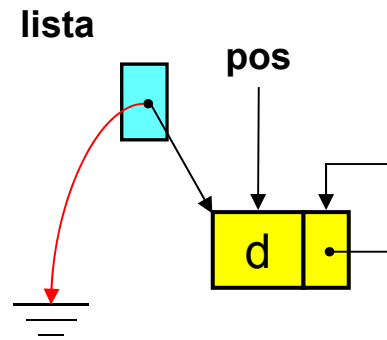
Eliminación
en cualquier posición,
excepto al principio

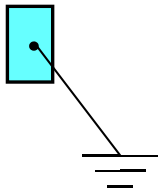
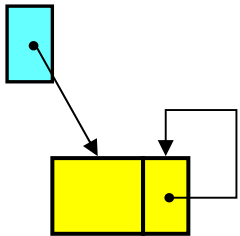
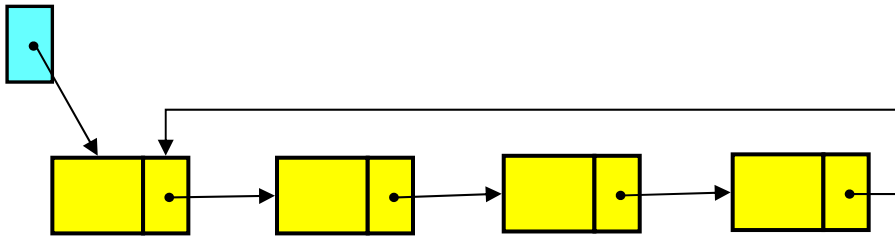


Eliminación al principio



Eliminación al principio





función: **elimina**

recibe: pos , $lista$

regresa: nada

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

Sí: terminar

¿ $pos \rightarrow sig = pos$?

Sí: $lista = \text{NULO}$

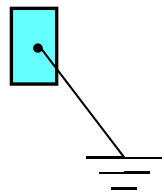
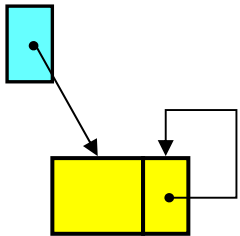
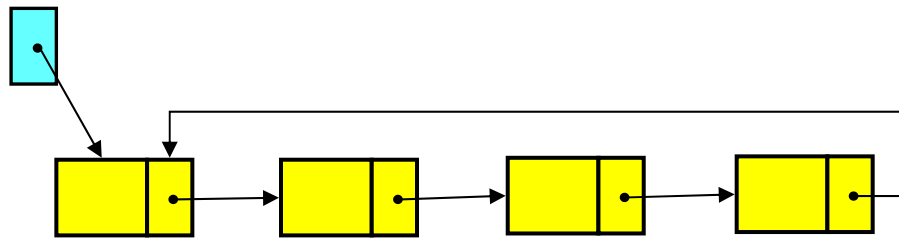
No: ¿ $pos = lista$?

Sí: **ultimo**($lista$) $\rightarrow sig = lista \rightarrow sig$

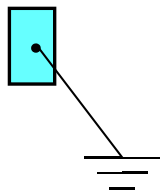
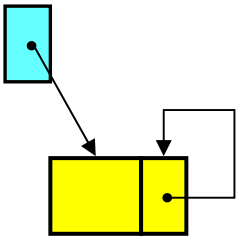
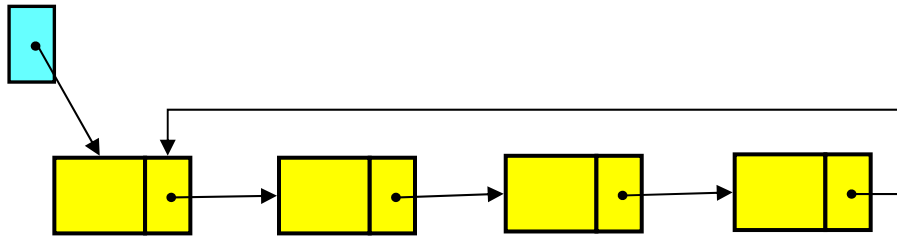
$lista = lista \rightarrow sig$

No: **anterior**(pos , $lista$) $\rightarrow sig = pos \rightarrow sig$

liberar espacio de memoria de pos



función: ***primero***
 recibe: *lista*
 regresa: posición
 regresar: *lista*



función: **último**

recibe: *lista*

regresa: posición

¿vacía(*lista*)?

Sí: regresar: NULO

terminar

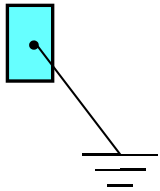
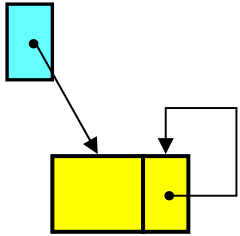
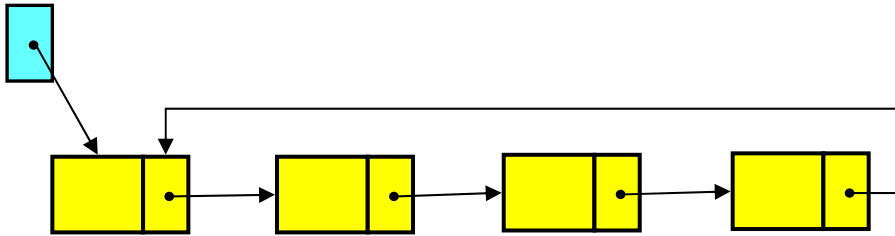
aux = *lista*

mientras *aux* → sig ≠ *lista*

aux = *aux* → sig

fin mientras

regresar: *aux*



función. **Anterior**

recibe: *pos*, *lista*

regresa: posición

¿vacía(*lista*) ó *pos* = NULO?

Sí: regresar: NULO

terminar

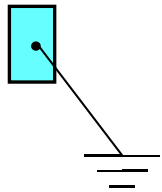
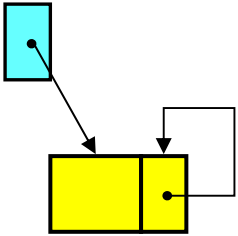
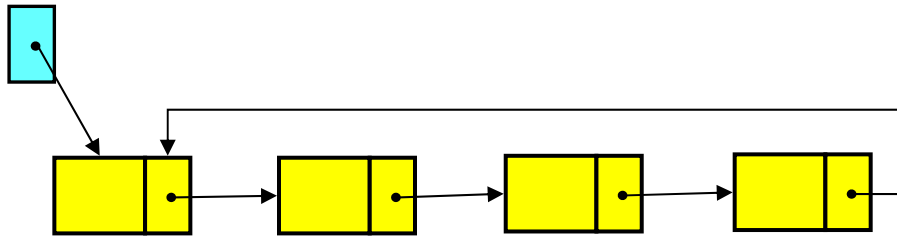
aux = *lista*

mientras *aux* → sig ≠ *pos*

aux = *aux* → sig

fin mientras

regresar: *aux*



función: **siguiente**

recibe: *pos*, *lista*

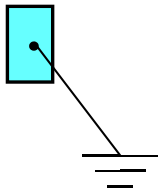
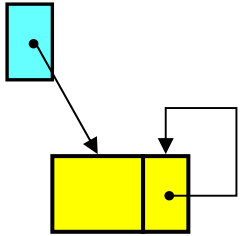
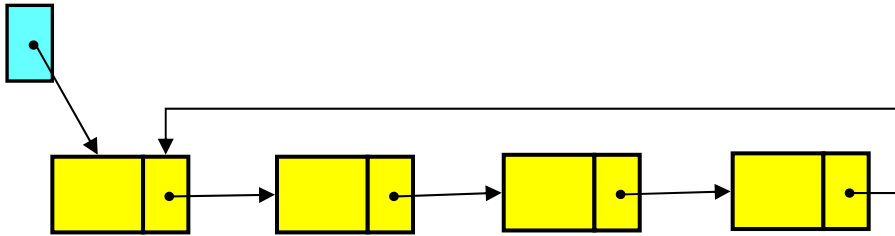
regresa: posición

¿vacía(*lista*) ó *pos* = NULO?

Sí: regresar: NULO

terminar

regresar: *pos* → sig



función: **localiza**

recibe: *elem*, *lista*

regresa: posición

¿**vacía**(*lista*)?

Sí: regresar: NULO

aux = *lista*

hacer:

¿*aux*→*elem* = *elem*?

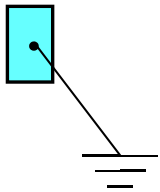
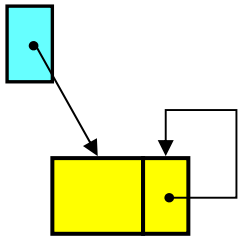
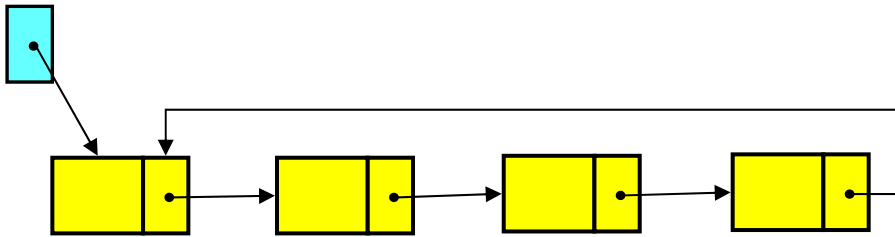
Sí: regresar: *aux*

salir

aux = *aux*→*sig*

mientras *aux* ≠ *lista*

regresar: NULL



función: **recupera**

recibe *pos*, *lista*

regresa: elemento

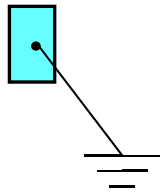
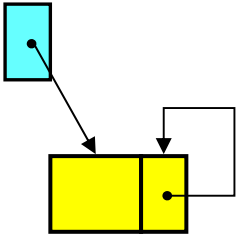
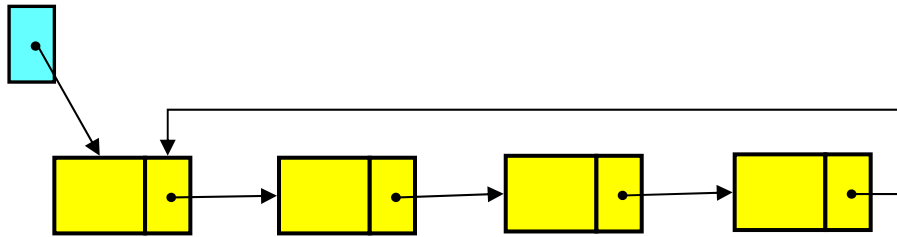
¿vacía(*lista*) ó *pos* = NULO?

Sí: ¡error de excepción!

Insuficiencia de datos
terminar



No: regresar: *pos* → elem



función: **imprime**

recibe: *lista*

regresa: nada

¿**vacía**(*lista*)?

Sí: terminar

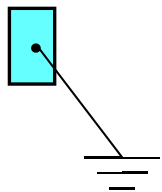
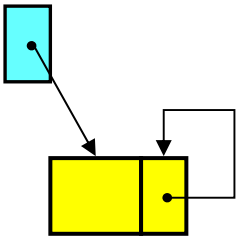
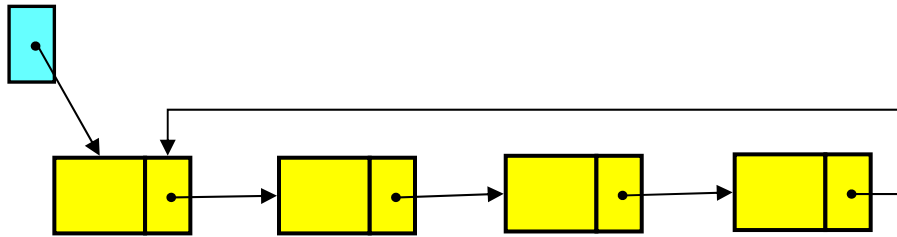
aux = *lista*

hacer:

imprimir *aux* → elem

aux = *aux* → sig

mientras *aux* ≠ *lista*



función: **anula**

recibe: *lista*

regresa: nada

¿**vacía**(*lista*)?

Sí: terminar

marca = *lista*

hacer:

aux = *lista*

lista = *lista* → sig

liberar espacio de memoria de *aux*

mientras *lista* ≠ *marca*

lista = NULO

