

# UNIDAD I. ESTRUCTURA DE DATOS BÁSICAS (LISTAS DOBLEMENTE LIGADAS)

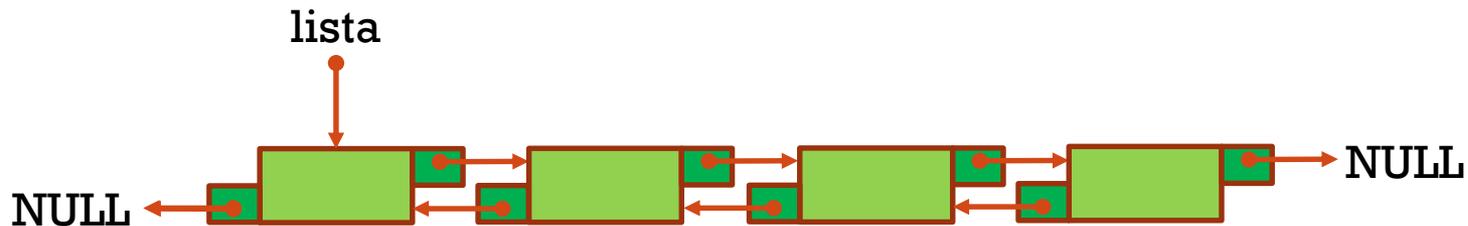
Francisco J. Hernández López

fcoj23@cimat.mx



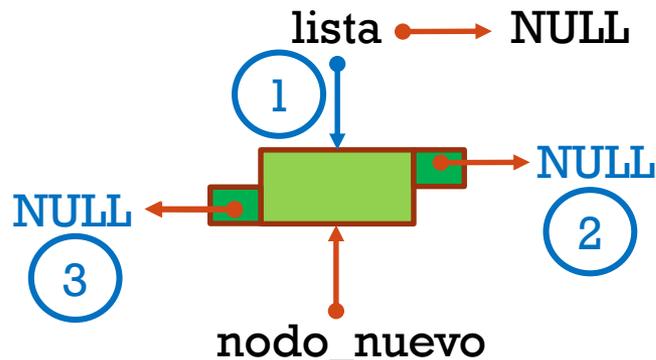
# LISTAS DOBLEMENTE LIGADAS

- Son listas ligadas en las que cada nodo tiene dos enlaces (o apuntables):
  - Uno apunta al nodo siguiente
  - El otro apunta al nodo anterior
- Se puede recorrer en ambos sentidos a partir de cualquier nodo



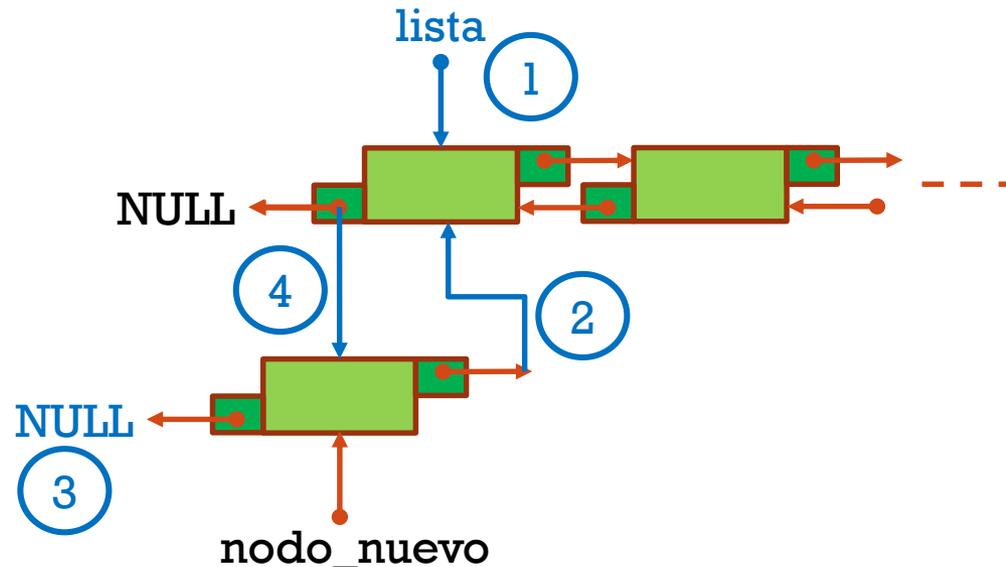
```
typedef struct _nodo{
    int valor;//Datos
    struct _nodo *anterior;//Apuntando al nodo anterior
    struct _nodo *siguiente;//Apuntando al nodo siguiente
}tipoNodo;
```

# AÑADIR UN ELEMENTO A UNA LISTA DOBLEMENTE LIGADA VACÍA



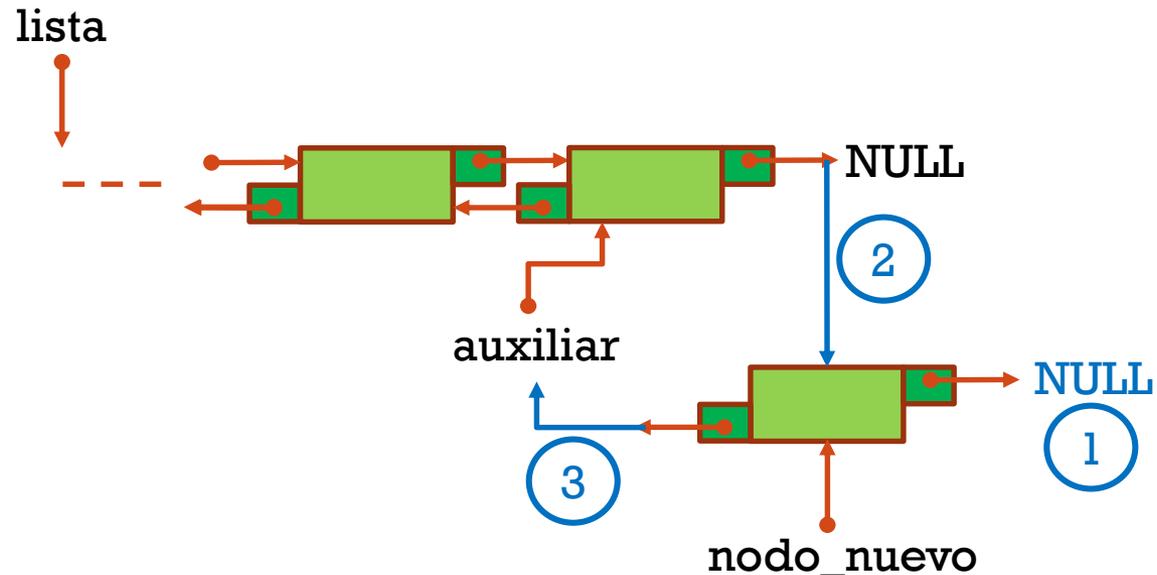
1. `lista = nodo_nuevo`
2. `lista→siguiente = NULL`
3. `lista→anterior = NULL`

# INSERTAR UN ELEMENTO EN LA PRIMERA POSICIÓN DE LA LISTA DOBLEMENTE LIGADA



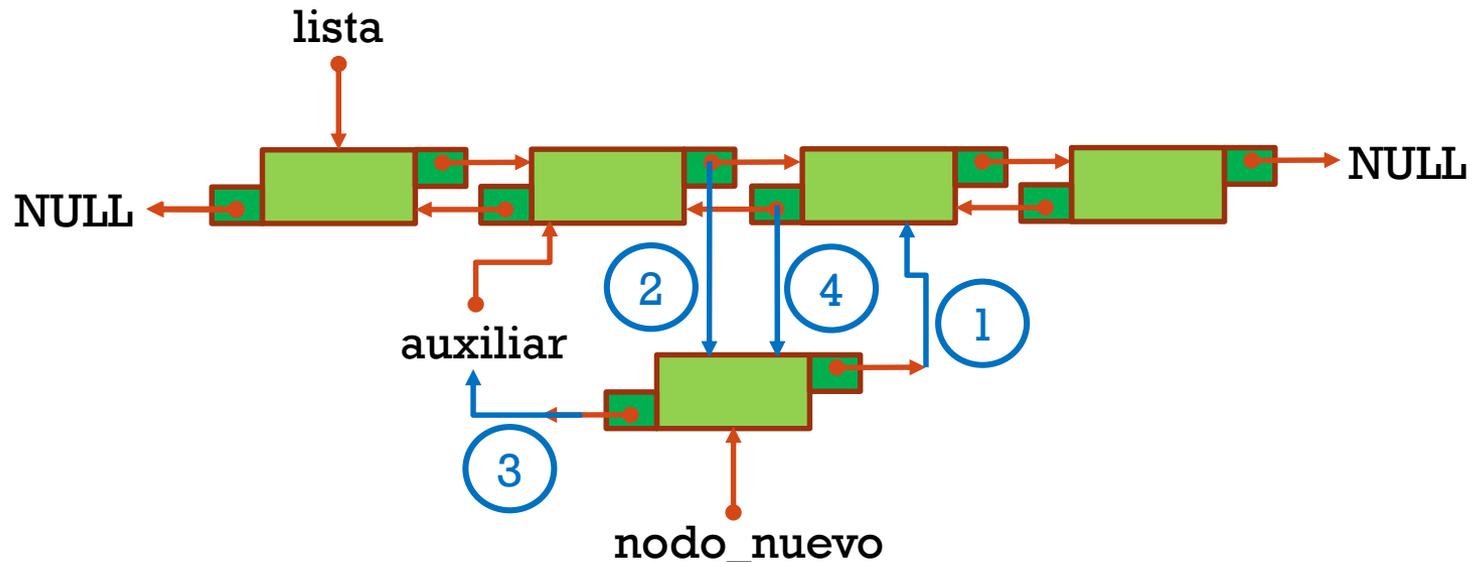
1. Hacemos que lista apunte al primer elemento de la lista
2. `nodo_nuevo` → siguiente = lista
3. `nodo_nuevo` → anterior = NULL (ya que lista puede apuntar a cualquier nodo, entonces: `nodo_nuevo` → anterior = lista → anterior)
4. lista → anterior = `nodo_nuevo`

# INSERTAR UN ELEMENTO EN LA ÚLTIMA POSICIÓN DE UNA LISTA DOBLEMENTE LIGADA



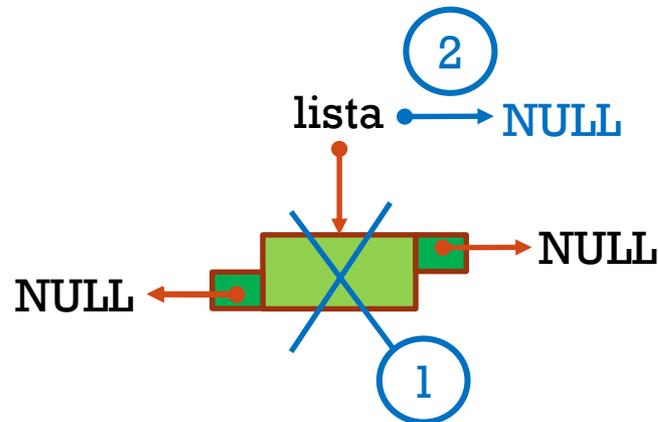
1. `nodo_nuevo` → siguiente = NULL
2. `auxiliar` → siguiente = `nodo_nuevo`
3. `nodo_nuevo` → anterior = `auxiliar`

# INSERTAR UN ELEMENTO A CONTINUACIÓN DE UN NODO CUALQUIERA



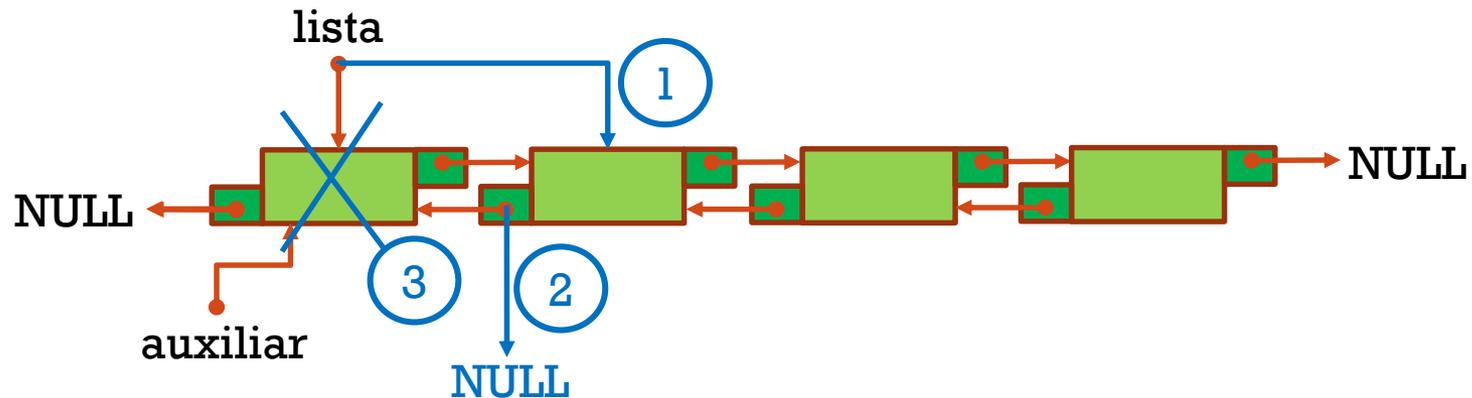
1.  $\text{nodo\_nuevo} \rightarrow \text{siguiente} = \text{auxiliar} \rightarrow \text{siguiente}$
2.  $\text{auxiliar} \rightarrow \text{siguiente} = \text{nodo\_nuevo}$
3.  $\text{nodo\_nuevo} \rightarrow \text{anterior} = \text{auxiliar}$
4.  $\text{nodo\_nuevo} \rightarrow \text{siguiente} \rightarrow \text{anterior} = \text{nodo\_nuevo}$

# ELIMINAR EL ÚNICO NODO EN UNA LISTA DOBLEMENTE LIGADA



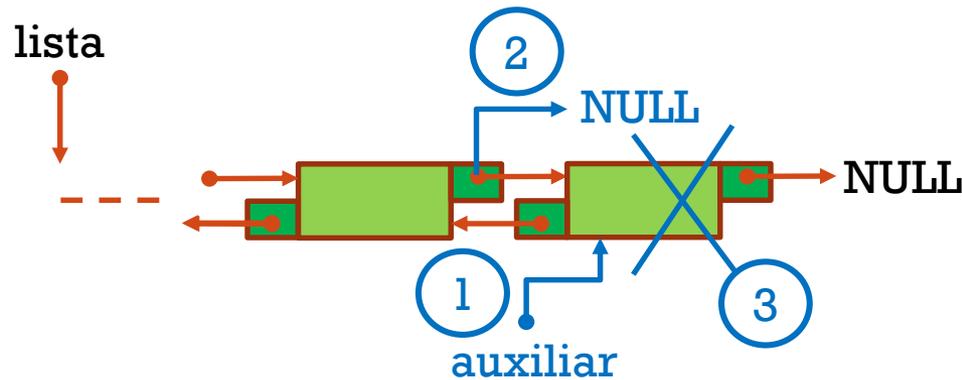
1. Liberar la memoria apuntada por lista
2. lista = NULL

# ELIMINAR EL PRIMER NODO DE LA LISTA DOBLEMENTE LIGADA



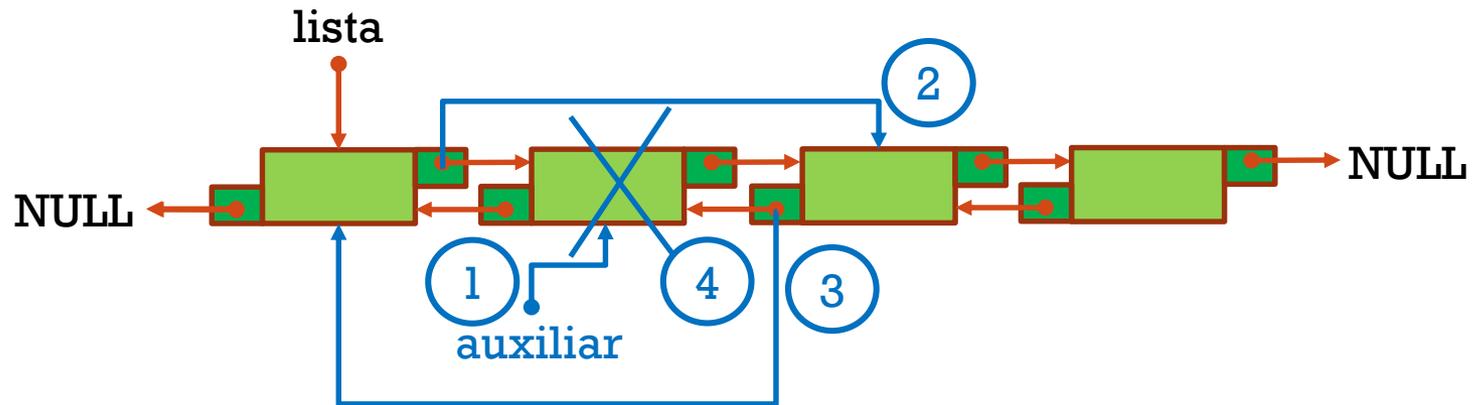
1. lista = lista → siguiente
2. auxiliar → siguiente → anterior = NULL
3. Eliminar la memoria apuntada por auxiliar

# ELIMINAR EL ÚLTIMO NODO DE UNA LISTA DOBLEMENTE LIGADA



1. Buscar el último nodo (“auxiliar” apunta al último nodo)
2. auxiliar → anterior → siguiente = NULL
3. Eliminar la memoria apuntada por auxiliar

# ELIMINAR UN NODO INTERMEDIO DE UNA LISTA DOBLEMENTE LIGADA



1. Buscar el nodo a eliminar (“auxiliar”)
  - a) Si “lista” apunta a “auxiliar” entonces
    - $lista = lista \rightarrow anterior$  ó  $lista = lista \rightarrow siguiente$
2.  $auxiliar \rightarrow anterior \rightarrow siguiente = auxiliar \rightarrow siguiente$
3.  $auxiliar \rightarrow siguiente \rightarrow anterior = auxiliar \rightarrow anterior$
4. Liberar memoria apuntada por auxiliar

# PROGRAMAR UNA LISTA DOBLEMENTE LIGADA