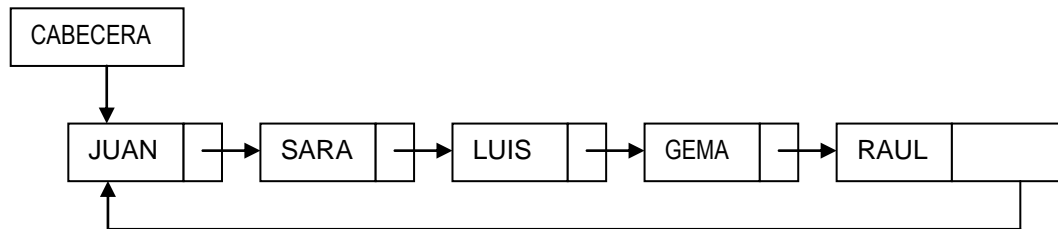


LISTAS CIRCULARES

Lista circular es una lista enlazada en la que el enlace del último nodo apunta al primer nodo.

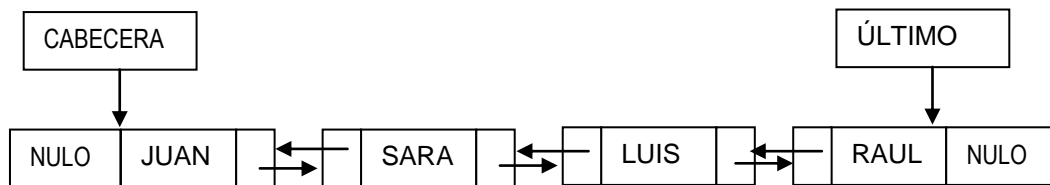
Con las listas circulares se tiene la ventaja de que cualquier nodo de la lista es accesible desde todos los demás, facilitando así algunas operaciones. Sin embargo, al recorrer listas circulares se corre el riesgo de caer en ciclos infinitos, si no se tiene el cuidado de implementar condiciones que marquen el final del recorrido. Una posible condición para identificar el final del recorrido, habiéndose iniciado este desde el inicio de la lista, es cuando el enlace del nodo es igual a la cabecera.



LISTAS DOBLEMENTE ENLAZADAS

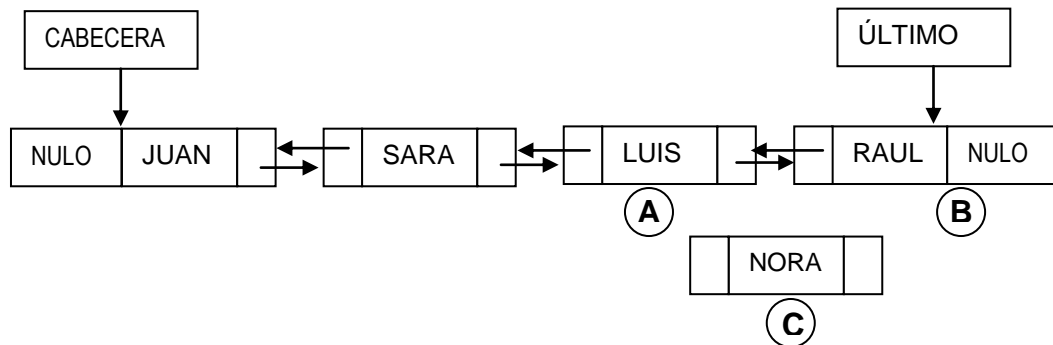
Las listas enlazadas o ligadas se pueden recorrer solamente en un sentido, de principio a fin. Las listas doblemente enlazadas ofrecen la flexibilidad de poderse recorrer en ambos sentidos, ya sea de principio a fin o de fin a principio.

Una lista doblemente enlazada es un conjunto de nodos en los que cada nodo contiene tres *campos*, el de información, el enlace al nodo *siguiente* y el enlace al nodo *anterior*. Para poder efectuar el recorrido de fin a inicio, se hace necesario contar con una variable adicional, equivalente a la cabecera, que apunta a la dirección del *último* nodo. La figura muestra un ejemplo de lista doblemente enlazada.



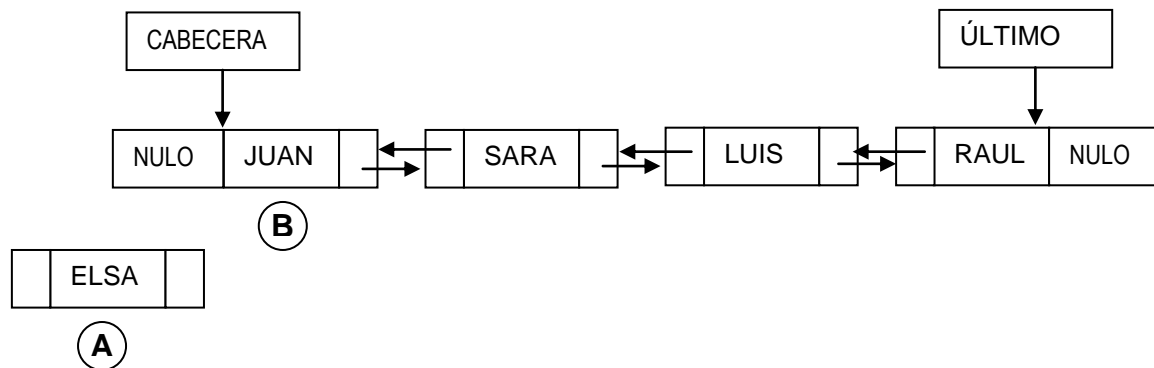
Insertar

En medio



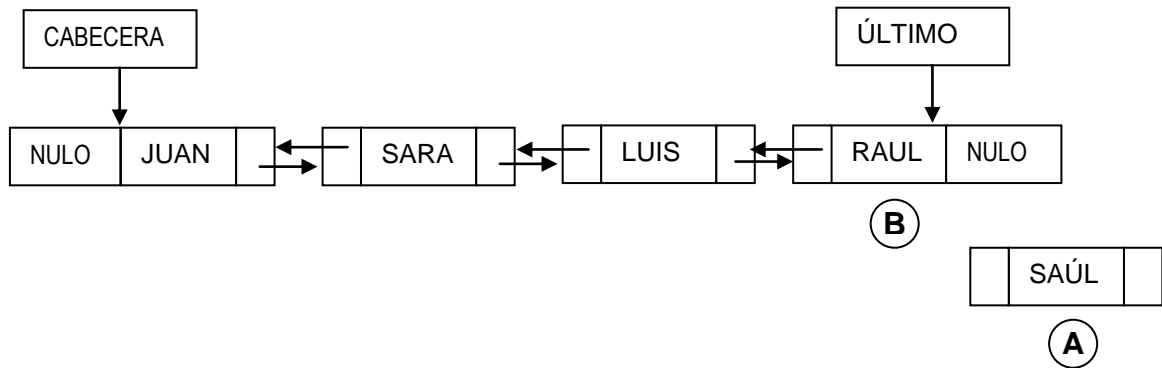
1. Localizar posición para insertar
2. Generar nuevo nodo "C"
3. Incluir información en el nodo "C"
4. Enlace siguiente de "C" apunta a "B"
5. Enlace anterior de "C" apunta a "A"
6. Enlace siguiente de "A" apunta a "C"
7. Enlace anterior de "B" apunta a "C"

Al principio



1. Generar nuevo nodo "A"
2. Incluir información en el nodo "A"
3. Enlace siguiente de "A" apunta a "B"
4. Enlace anterior de "A" apunta a "nulo"
5. Enlace anterior de "B" apunta a "A"
6. Cabecera apunta a "A"

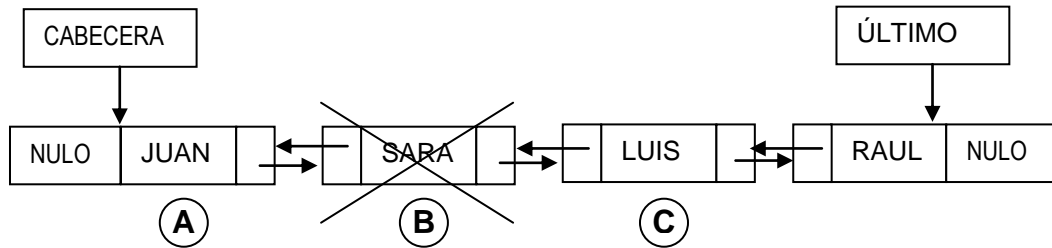
Al final



1. Generar nuevo nodo "A"
2. Incluir información en el nodo "A"
3. Enlace siguiente de "A" apunta a "nulo"
4. Enlace anterior de "A" apunta a "B"
5. Enlace siguiente de "B" apunta a "A"
6. Último apunta a "A"

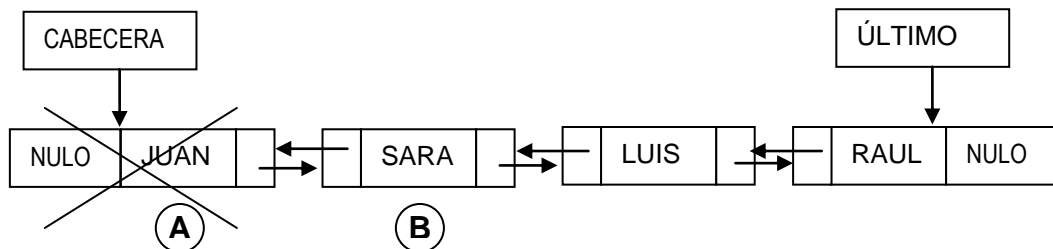
Extraer

En medio



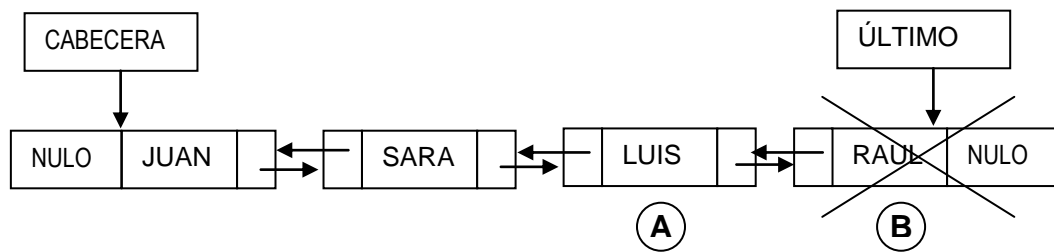
1. Localizar nodo a borrar "B"
2. Enlace siguiente de "A" apunta a "C"
3. Enlace anterior de "C" apunta a "A"
4. Se pone disponible "B"

Al principio



1. Cabecera apunta a "B"
2. Enlace anterior de "B" apunta a "nulo"
3. Se pone disponible "A"

Al final



1. Último apunta a "A"
2. Enlace siguiente de "A" apunta a "nulo"
3. Se pone disponible "B"