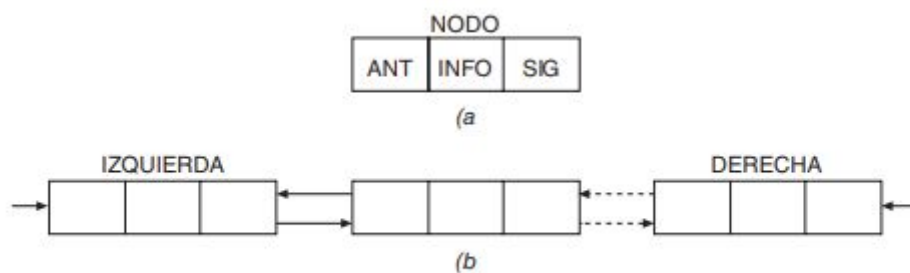
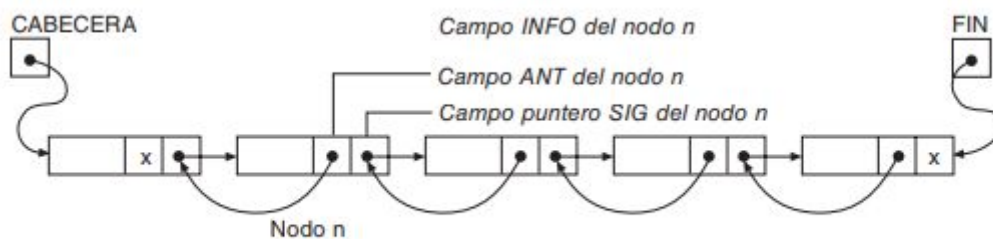


## LISTAS DOBLEMENTE ENLAZADAS

En las listas lineales estudiadas anteriormente el recorrido de ellas sólo podía hacerse en un único sentido: de izquierda a derecha (principio a final). En numerosas ocasiones se necesita recorrer las listas en ambas direcciones. Las listas que pueden recorrerse en ambas direcciones se denominan listas doblemente enlazadas. En estas listas cada nodo consta del campo INFO de datos y dos campos de enlace o punteros: ANTERIOR(ANT) y SIGUIENTE(SIG) que apuntan hacia adelante y hacia atrás. Como cada elemento tiene dos punteros, una lista doblemente enlazada ocupa más espacio en memoria que una lista simplemente enlazada para una misma cantidad de información. La lista necesita dos punteros CABECERA y FIN2 que apuntan hacia el primero y último nodo. La variable CABECERA y el puntero SIG permiten recorrer la lista en el sentido normal y la variable FIN y el puntero ANT permiten recorrerla en sentido inverso.



**Figura 12.9.** Lista doblemente enlazada.

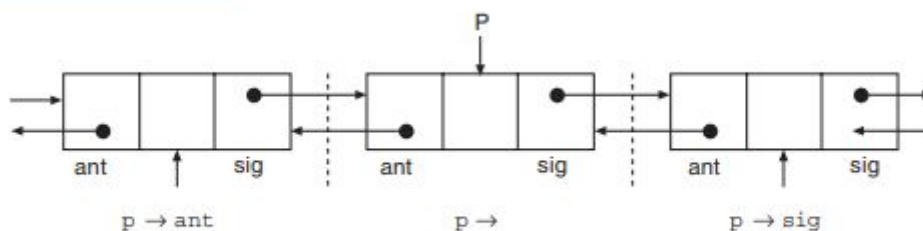


**Figura 12.10.** Lista doble.

Una propiedad fundamental de las listas doblemente enlazadas es que para cualquier puntero P de la lista:

$\text{nodo}[\text{nodo}[p].\text{sig}].\text{ant} = p$

$\text{nodo}[\text{nodo}[p].\text{ant}].\text{sig} = p$

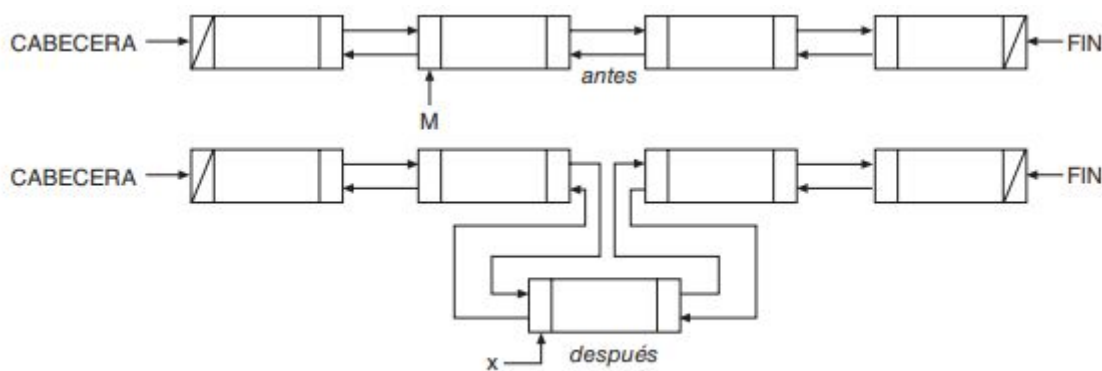


**Figura 12.11.**

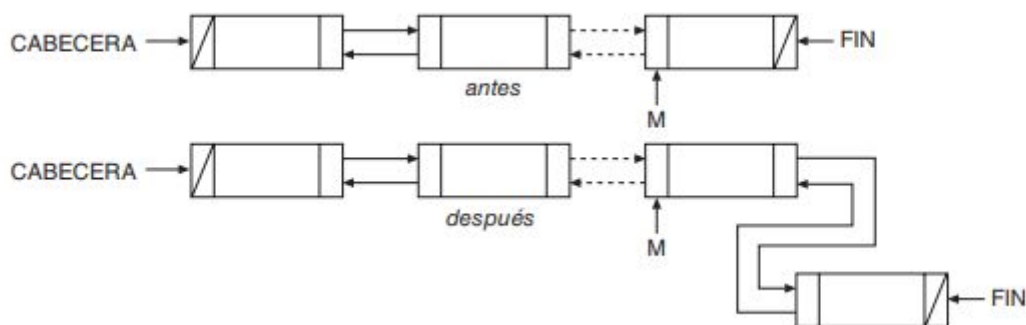
Insertión

La inserción de un nodo a la derecha de un nodo especificado, cuya dirección está dada por la variable M, puede presentar varios casos:

1. La lista está vacía; se indica mediante  $M = \text{NULO}$  y CABECERA y FIN son también NULO. Una inserción indica que CABECERA se debe fijar con la dirección del nuevo nodo y los campos ANT y SIG también se establecen en NULO.
2. Insertar dentro de la lista: existe un elemento anterior y otro posterior de nuevo nodo.
3. Insertar a la derecha del nodo del fin de la lista. Se requiere que el apuntador FIN sea modificado.

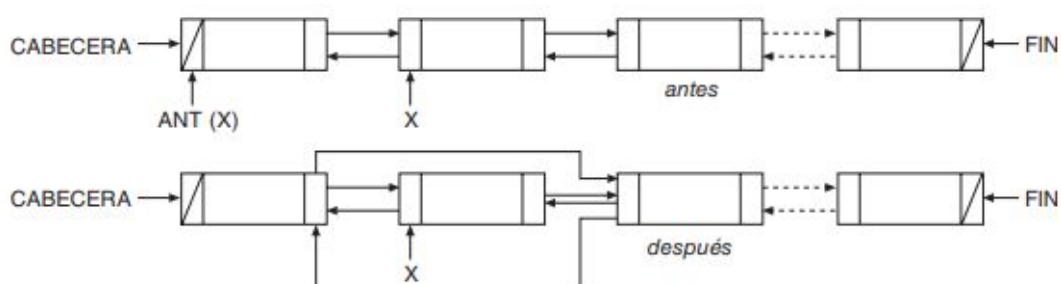


**Figura 12.12.** Inserción en una lista doblemente enlazada.



### Eliminación

La operación de eliminación es directa. Si la lista tiene un simple nodo, entonces los punteros de los extremos izquierdo y derecho asociados a la lista se deben fijar en NULO. Si el nodo del extremo derecho de la lista es el señalado para la eliminación, la variable FIN debe modificarse para señalar el predecesor del nodo que se va a borrar de la lista. Si el nodo del extremo izquierdo de la lista es el que se desea borrar, la variable CABECERA debe modificarse para señalar el elemento siguiente. La eliminación se puede realizar dentro de la lista



**Figura 12.14.** Eliminación de un nodo X en una lista doblemente enlazada.

Bibliografia:

Luis Joyanes Aguilar, Ignacio Zahonero Martinez. (2008). Estructura de datos en Java.  
Mexico: MCGRAW-HILL.