

Cuaderno de ejercicios – Listas enlazadas

1. Implemente una función **recursiva** que permita eliminar un Nodo a partir del valor que almacena.

Nodo eliminarNodo(Nodo lista, int eliminar)

1. Implemente una función que permita agregar un Nodo a una lista, pero siempre indicando la posición en donde debe ir.

Obs.: Tenga presente que si la posición es **inferior** a 0, el Nodo debe ser agregado en la posición 0, en el caso que sea **superior** a N, el nodo debe ser agregado en la posición N.

Nodo agregarNodoEn(Nodo lista, int valor, int posicion)

Ejemplo

10 -> 21 -> 7 -> 25 -> 78 -> NULL

Agregar 50 en la posición 3.

Resultado que debería ser: 10 -> 21 -> 7 -> **50** -> 25 -> 78 -> NULL

Agregar 50 en la posición 1.

Resultado que debería ser: 10 -> **50** -> 21 -> 7 -> 25 -> 78 -> NULL

Agregar 50 en la posición 0.

Resultado que debería ser: **50** -> 10 -> 21 -> 7 -> 25 -> 78 -> NULL

Agregar 50 en la posición -5.

Resultado que debería ser: **50** -> 10 -> 21 -> 7 -> 25 -> 78 -> NULL

Agregar 50 en la posición 100.

Resultado que debería ser: 10 -> 21 -> 7 -> 25 -> 78 -> **50** -> NULL

Importante

1. Para cada función realizar un par de ejemplos de ejecución.
2. Todos los ejercicios deben ser ejecutados en un único Main.
3. Código comentado no se tomará en cuenta.
4. No es posible modificar los prototipos de las funciones indicadas.
5. Subir a Moodle un **único archivo .c** con los ejercicios resueltos.
6. Si el archivo .c tiene **otra extensión o no compila**, se asignará la nota mínima (1).