

# Introducción a las listas enlazadas y doblemente enlazadas

## Conceptos y Operaciones Básicas

### Listas Enlazadas Ordenadas

Las listas enlazadas son una estructura de datos lineal donde cada elemento, conocido como nodo, contiene un valor y un puntero al siguiente nodo de la secuencia. En una lista enlazada ordenada, los nodos están organizados en orden ascendente o descendente, según un criterio específico.

#### Inserción

Para insertar un nuevo nodo en una lista enlazada ordenada, se debe encontrar la posición correcta para mantener el orden. El nuevo nodo se inserta ajustando los punteros de los nodos adyacentes.

Diagrama de Inserción:

[10] -> [20] -> [30]

Insertar 25:

[10] -> [20] -> [25] -> [30]

#### Borrado

Para borrar un nodo de una lista enlazada ordenada, se debe ajustar el puntero del nodo anterior al nodo que se va a eliminar.

Diagrama de Borrado:

[10] -> [20] -> [30]

Borrar 20:

[10] -> [30]

#### Búsqueda

La búsqueda en una lista enlazada ordenada implica recorrer la lista hasta encontrar el nodo deseado, comparando cada valor con el valor buscado.

LEAD University  
Tarea Programada 3  
Diagrama de Búsqueda:

[10] -> [20] -> [30]

Buscar 20:

[10] -> [20] (encontrado)

## Listas Doblemente Enlazadas Ordenadas

Las listas doblemente enlazadas son una extensión de las listas enlazadas donde cada nodo tiene dos punteros: uno al siguiente nodo y otro al nodo anterior. En una lista doblemente enlazada ordenada, los nodos también están organizados en orden ascendente o descendente.

### Inserción

La inserción en una lista doblemente enlazada ordenada requiere ajustar tanto los punteros del nodo anterior como del siguiente para mantener el orden.

Diagrama de Inserción:

[10] [20] [30]

Insertar 25:

[10] [20] [25] [30]

### Borrado

Para borrar un nodo en una lista doblemente enlazada ordenada, se deben ajustar los punteros de los nodos anterior y siguiente al nodo que se va a eliminar.

Diagrama de Borrado:

[10] [20] [30]

Borrar 20:

[10] [30]

### Búsqueda

La búsqueda en una lista doblemente enlazada ordenada es similar a la de una lista enlazada, pero se puede recorrer en ambas direcciones (adelante y atrás).

Diagrama de Búsqueda:

LEAD University  
Tarea Programada 3

[10] [20] [30]

Buscar 20:

[10] [20] (encontrado)

Puede investigar mas en el siguiente link:

[https://www.tutorialspoint.com/data\\_structures\\_algorithms/linked\\_list\\_algorithms.htm](https://www.tutorialspoint.com/data_structures_algorithms/linked_list_algorithms.htm)

## ENUNCIADO PARTE 1

---

Usando como base el código proporcionado, debe crear las clases nodo (¿o mantenerla?), lista\_enlazada\_ordenada y lista\_doble\_ordenada. Además debe crear la clase persona.

La clase persona tendrá por campos nombre, apellido1, apellido2, edad.

Con lista\_enlazada\_ordenada y lista\_doble\_ordenada debe programar los procesos de inserción, imprimir, búsqueda y borrado en la lista.

La inserción en ambas listas (la normal y la doble\_ordenada) deberá ser ORDENADA, por la edad de las personas, de menor a mayor.

La función imprimir me diga la cantidad de total de personas en la lista y las imprimirá.

La función búsqueda buscare personas:

- Por edad: pedirá la edad y para la lista sencilla buscare desde la cabeza/inicio hacia el final. En el caso de la lista doblemente enlazada, revisará la cabeza y la cola, determinara cual es el campo con menor diferencia respecto a la edad buscada y arrancara de él. Si no encuentra a nadie, dira que nadie tiene esa edad. Si encuentra una o mas personas con la edad dada, listara todas.
- Por aproximación sobre el nombre ("contains" del nombre)
- Por aproximación sobre CUALQUIERA de los apellidos ("contains" del apellido) y listará todas las personas que hacen match con el criterio.
- Borrado: borrará por posición: deberá pedirle el # de item de la lista al usuario. Si existe tal ítem, lo borra y notifica. Si no, avisa que la lista tiene "X" elementos.

TOME EN CUENTA QUE EL CODIGO QUE PREGUNTA AL USUARIO POR EL TIPO DE BUSQUEDA Y LOS PARAMETROS, NO DEBE ESTAR EN LA CLASE DE LA LISTA O LA CLASE DEL NODO.

## REQUERIMIENTOS DE LA PARTE 1

---

Debe crear un programa SEPARADO para ambos tipos de lista. Cada programa mostrara un menú con las siguientes opciones:

- Agregar persona a la lista
- Lisar personas

- Borrar personas
- Buscar personas (esta opción ofrecerá las opciones de búsqueda dadas anteriormente).

## REQUERIMIENTOS PARTE 2

---

Usando la estructura de lista simple enlazada (o sea, como base una estructura parecida a lo implementado en lista\_enlazada\_ordenada) debe implementar una COLA DE PRIORIDAD de personas.

La lista NO estará ordenada por edad, sino por orden de llegada (como corresponde a una cola) pero si la persona es mayor de 65 años:

- Si no hay nadie de 65 o mas en el inicio de la cola, lo coloca al inicio.
- Si ya hay una o mas personas con mas de 65 años al inicio de la cola, lo coloca después de la última persona que haya llegado antes de 65 años

Ejemplo (simplificado para mostrar el proceso **NO ES QUE DEBAN IMPRIMIR ASI LA COLA EN ESTA PARTE O LA PRIMERA**)

[] cola vacia al inicio

Llega una persona de 20 años (p20)

[P20]

Llega una persona de 45 años (p45)

[P20 P45]

Llega una persona de 35 (p35)

[P20 P45 P35]

Sale una persona de la cola

[P45 P35]

Llega una persona de 66 años (p66)

[P66 P45 P35]

Llega una persona de 19 años (p19)

[P66 P45 P19]

LEAD University

Tarea Programada 3

Llega una persona de 65 (p65)

[P66 P65 P45 P19]

Llega una persona de 18 años

[P66 P65 P45 P19 P18]

Es decir, la cola solo tiene una implementación diferenciada a la hora de insertar.

ENTREGA

---

17 de Marzo del 2025 antes de la media noche.