

## **Problema 1 — Invertir una Lista Enlazada Explicación**

sencilla (en tus propias palabras)

El algoritmo invierte la lista cambiando la dirección de los punteros. Para no perder la referencia al resto de la lista se usan tres variables:

anterior: guarda la parte ya invertida

actual: nodo que estoy recorriendo

siguiente: guarda temporalmente el resto

Se repite hasta que la lista se termina.

Método dentro de ListaEnlazada

Agrega esto a tu clase:

```
invertir() {  
  
    let anterior = null;  
    let actual = this.cabeza;  
    let siguiente = null;  
  
    // Recorremos la lista  
    while (actual !== null) {
```

```
// Guardar el siguiente nodo  
siguiente = actual.siguinte;
```

```
// Invertir el puntero  
actual.siguinte = anterior;
```

```
// Avanzar punteros  
anterior = actual;  
actual = siguiente;  
}
```

```
// Actualizar la cabeza this.cabeza  
= anterior;  
}
```

Prueba del método

```
const lista = new ListaEnlazada();
```

```
lista.insertarFinal("A");  
lista.insertarFinal("B");  
lista.insertarFinal("C");  
lista.insertarFinal("D");
```

```
console.log("Lista original:");
lista.imprimir();
// A -> B -> C -> D -> null
```

```
lista.invertir();
```

```
console.log("Lista invertida:");
lista.imprimir();
// D -> C -> B -> A -> null
```

## Problema 2 — Encontrar el Nodo Medio

Explicación sencilla

Se usan dos punteros:

Uno avanza de 1 en 1

Otro avanza de 2 en 2

Cuando el rápido llega al final, el lento queda en el centro.

Esto evita contar nodos.

Método

```
encontrarMedio() {
```

```
if (this.cabeza === null) return null;

let lento = this.cabeza; let
rapido = this.cabeza;

while (rapido !== null && rapido.siguiente !== null) {
    lento = lento.siguiente;
    rapido = rapido.siguiente.siguiente;
}

return lento.dato;
}

Prueba
const lista2 = new ListaEnlazada();

lista2.insertarFinal(10);
lista2.insertarFinal(20);
lista2.insertarFinal(30);
lista2.insertarFinal(40);
lista2.insertarFinal(50);
```

```
lista2.imprimir();
// 10 -> 20 -> 30 -> 40 -> 50 -> null

console.log("Nodo medio:");
console.log(lista2.encontrarMedio());
// 30
```

Explica  
ción  
Inverti  
r lista

La lista se invierte cambiando la dirección de cada enlace.

Se utilizan tres punteros para no perder la referencia a los nodos restantes.

### Encontrar el medio

Se usan dos punteros que avanzan a distinta velocidad.

Cuando el puntero rápido termina el recorrido, el lento queda en el nodo central.

Complejidad General

Algoritmo	Tiempo	Espacio
Invertir lista	$O(n)$	$O(1)$
Encontrar medio	$O(n)$	$O(1)$