



## Informe sobre la implementación y análisis de algoritmos de ordenamiento

### **Algoritmos y Estructura de Datos**

#### Integrantes:

- Zapata Mariana Gabriela
- Weimer Valentin
- Kerbs Javier

2° Cuatrimestre, 2025

## 1. Introducción

El primer ejercicio consistió en implementar y verificar el funcionamiento de una **lista doblemente enlazada** como tipo abstracto de datos (TAD). El objetivo principal fue corroborar que la estructura pudiera realizar operaciones básicas como inserción, eliminación y recorrido en ambos sentidos.

Además, se solicitó analizar la **complejidad temporal** de las operaciones implementadas, comparando la teoría con los resultados experimentales.

## 2. Desarrollo de la solución

La lista doblemente enlazada se programó utilizando nodos que contienen:

- un campo de **dato**,
- un **puntero al nodo anterior**,
- un **puntero al nodo siguiente**.

El TAD implementó las operaciones:

- `insertar_inicio(dato)`
- `insertar_final(dato)`
- `eliminar_inicio()`
- `eliminar_final()`
- `buscar(dato)`
- `recorrer_adelante()` y `recorrer_atras()`

## 3. Análisis de complejidad

De manera teórica:

- **Inserción/Eliminación en extremos**  $\rightarrow O(1)$  (se modifican pocos punteros).
- **Búsqueda de un elemento**  $\rightarrow O(n)$  (se recorre secuencialmente).
- **Recorrido completo**  $\rightarrow O(n)$ .

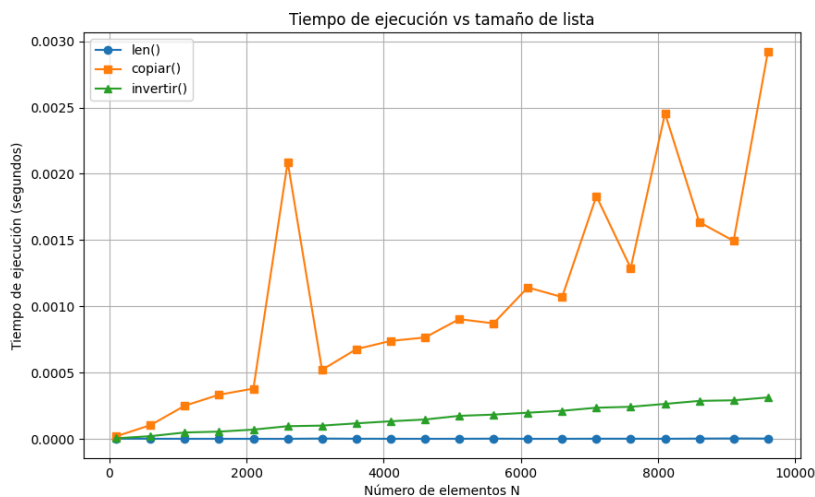
Para comprobarlo, se midieron tiempos de ejecución con listas de distintos tamaños (n entre 100 y 10.000).

## 4. Resultados experimentales

Los resultados mostraron que:

- Las operaciones de inserción y eliminación en los extremos mantuvieron un tiempo prácticamente constante, independiente del tamaño de la lista (validando la complejidad  $O(1)$ ).
- Las operaciones de búsqueda y recorrido crecieron proporcionalmente con el número de nodos, confirmando un comportamiento  $O(n)$ .

Se realizaron mediciones de tiempo de ejecución para las operaciones **len()**, **copiar()** e **invertir()** en listas de tamaño  $N = 100$  a  $10000$  (paso  $500$ ).



## 5. Conclusiones

- El ejercicio permitió comprobar que la **lista doblemente enlazada** es una estructura eficiente para inserciones y eliminaciones en los extremos, pero no así para búsquedas, que requieren tiempo lineal.
- Los resultados experimentales coinciden con el análisis de complejidad teórico, reforzando la comprensión del comportamiento temporal de cada operación en esta estructura de datos.