

Evidencia 2

Ventajas:

Inserción y eliminación eficiente: Se pueden realizar en tiempo constante si se tiene acceso al nodo correcto. Permite recorrer la lista en ambas direcciones, útil para operaciones específicas.

Mejor rendimiento con quicksort: Quicksort optimiza el tiempo de ordenamiento con su complejidad de $O(n \log n)$, a diferencia de métodos menos eficientes como bubble sort el cual se tardaba mucho en ejecutar debido a su complejidad de $O(n^2)$

Desventajas:

Mayor consumo de memoria: Cada nodo almacena dos punteros adicionales, lo que aumenta el uso de memoria.

Acceso lineal: A diferencia de un arreglo o vector, donde se puede acceder directamente a cualquier elemento por su índice, en una lista doblemente enlazada hay que recorrerla para acceder a nodos específicos, resultando en un tiempo de acceso lineal.

Reflexión Metodo de Ordenamiento

Utilizamos el algoritmo de quicksort debido a que el tiempo de complejidad es mas eficiente para poder ordenar las IPs. Este nos dio un tiempo considerablemente bajo debido a que el archivo a analizar traía mucha información. Un algoritmo como bubble sort no es lo mas optimo para este caso ya que en su peor caso su complejidad es $O(n^2)$ y debido a esto, se debian ejecutar demasiadas operaciones por que una lista doblemente enlazada y no se puede acceder directamente a un índice sin iterar por los demas.