

UNIVERSIDAD INTERNACIONAL DEL ECUADOR

SEDE DE LOJA

Escuela de Ingeniería en Tecnologías de la Información

Diagrama de memoria: arreglo contiguo vs nodos/enlaces

Camila Doménica Loarte Aguilar

Estructura de Datos

Ing. Charlie Cárdenas

04 de noviembre de 2025

Algoritmo	Operaciones Principales	Fórmula	Big-O Tiempo	Big-O Espacio	Justificación
Comparar Pares	Bucle externo (n veces), Bucle interno (aprox. n veces), comparación (aprox. $n^2/2$)	$\frac{n(n - 1)}{2}$	$O(n^2)$	$O(1)$	Compara cada elemento con todos los demás, lo que requiere dos bucles anidados y resulta en un crecimiento cuadrático.
Encontrar Mínimo	Bucle for ($n-1$ veces), comparación if ($n-1$ veces), asignación (0 a $n-1$ veces).	$2n-2$	$O(n)$	$O(1)$	Recorre toda la lista (n elementos) una sola vez para encontrar el mínimo, usando solo una variable para el mínimo actual.
Contar Elementos	Bucle for (n veces), comparación if (n veces), incremento (0 a n veces), return (1 vez).	$2n$	$O(n)$	$O(1)$	Debe revisar toda la lista de n elementos para asegurar que cuenta todas las ocurrencias, usando solo una variable acumuladora.
Operación Constante	Acceso a elemento (2 veces), comparación if (1 vez), console.log (2 veces), return (1 vez).	5	$O(1)$	$O(1)$	Realiza un número fijo de operaciones independientemente del tamaño n de la lista (solo accede a los extremos).
Duplicar Elementos	Bucle for (n veces), multiplicación (n veces), push (n veces), return (1 vez).	$3n + 1$	$O(n)$	$O(n)$	Recorre todos los n elementos (tiempo $O(n)$) y crea una nueva lista de tamaño n (espacio $O(n)$) para el resultado.

Suma de Matriz	Bucle externo (N veces), Bucle interno (M veces), suma (N \times M veces). (Si es n x n: n^2 sumas).	$n^2 + 2n$	$O(n^2)$	$O(1)$	Utiliza bucles anidados que visitan cada uno de los $n \times n$ elementos de la matriz, usando solo una variable acumuladora.
Búsqueda Binaria	Bucle while ($\log 2(n)$ veces), comparación if ($\log_2(n)$ veces), operaciones aritméticas (constantes).	$\log_2(n)$	$O(\log(n))$	$O(1)$	Divide repetidamente la lista (ordenada) por la mitad, reduciendo el espacio de búsqueda a la mitad en cada paso.