
Cota superior de complejidad temporal en peor caso por estructura

Operación	Lista Enlazada	Lista Enlazada Ordenada	ABB (BST)	AVL	Heap	Trie
Pertenece	$O(n)$	$O(n)$	$O(n)$	$O(\log n)$	$O(n)$	$O(k)$
Inserción	$O(1)^\dagger$	$O(1)^\dagger$	$O(n)$	$O(\log n)$	$O(\log n)$	$O(k)$
Borrado	$O(1)^\dagger$	$O(1)^\dagger$	$O(n)$	$O(\log n)$	$O(n)$	$O(k)$
Buscar Mínimo	$O(n)$	$O(1)^\ddagger$	$O(n)$	$O(\log n)$	$O(1)^\ddagger$	$O(n)$
Borrar Mínimo	$O(n)$	$O(1)^{\dagger\ddagger}$	$O(n)$	$O(\log n)$	$O(\log n)^\ddagger$	$O(k)$

Siendo n la cantidad de elementos presentes en la colección. Y, en el caso del trie, siendo $|k|$ el largo de la clave (*key*).

[†]Considerando que, dado un iterador (puntero) al elemento, la inserción/remoción como tal –es decir, reacomodar los punteros de la estructura– se realiza en $O(1)$. Buscar el elemento es $O(n)$.

[‡]Considerando que la estructura preserva el orden total de *menor o igual*. ($\bullet \leq \bullet$). Es decir, dados a, b elementos pertenecientes a la estructura, a aparece antes que b si y sólo si $a < b$. Si $a = b$ entonces b podría aparecer antes que a .

En caso de la lista enlazada, significa que está ordenada de forma ascendente. En cambio, si se trata de un heap, entonces es un *min*-heap.