Data Structure time complexity

Data Structure	Time Complexity				lexity
	Average				
	Access	Search			
Array (http://en.wikipedia.org/wiki/Array_data_structure)	Θ(1)	Θ(n)			
Stack (stack-in-details.html)	Θ(n)	Θ(n)			
Queue (queue-in-details.html)	Θ(n)	Θ(n)			
Singly-Linked List (linked-list-in-details.html)	Θ(n)	Θ(n)			
Doubly-Linked List (http://en.wikipedia.org/wiki/Doubly_linked_list)	Θ(n)	Θ(n)			
Skip List (http://en.wikipedia.org/wiki/Skip_list)	`θ(log(n))`	`θ(log(n))`	`θ(log(n))`	`θ(log(n))`	O(n log(n))
Hash Table (http://en.wikipedia.org/wiki/Hash_table)	`N/A`	Θ(1)	Θ(1)	Θ(1)	O(n)
Binary Search Tree (http://en.wikipedia.org/wiki/Binary_search_tree)	`θ(log(n))`	`θ(log(n))`	`θ(log(n))`	`θ(log(n))`	O(n)
Cartesian Tree (https://en.wikipedia.org/wiki/Cartesian_tree)	`N/A`	`θ(log(n))`	`θ(log(n))`	`θ(log(n))`	O(n)
B-Tree (http://en.wikipedia.org/wiki/B_tree)	`θ(log(n))`	`θ(log(n))`	`θ(log(n))`	`θ(log(n))`	O(n)

Red-Black Tree (http://en.wikipedia.org/wiki/Red-black_tree)	`θ(log(n))`	`θ(log(n))`
AVL Tree (http://en.wikipedia.org/wiki/AVL_tree)	`θ(log(n))`	`θ(log(n))`

Array Sorting Algorithms

Algorithm	**Time Complexity**			
	Best	**Average**		
Quicksort (http://en.wikipedia.org/wiki/Quicksort)	Ω(n log(n))	Θ(n log(n))	`O(n^2)`	`O(log(n))`
Mergesort (http://en.wikipedia.org/wiki/Merge_sort)	Ω(n log(n))	Θ(n log(n))	O(n log(n))	O(n)
Heapsort (http://en.wikipedia.org/wiki/Heapsort)	Ω(n log(n))	Θ(n log(n))	O(n log(n))	O(1)
Bubble Sort (http://en.wikipedia.org/wiki/Bubble_sort)	Ω(n)	`θ(n^2)`	`O(n^2)`	O(1)
Insertion Sort (http://en.wikipedia.org/wiki/Insertion_sort)	Ω(n)	`θ(n^2)`	`O(n^2)`	O(1)

Selection Sort (http://en.wikipedia.org/wiki/Selection_sort)	Ω(n^2)	θ(n^2)	O(n^2)	O(1)
Tree Sort (https://en.wikipedia.org/wiki/Tree_sort)	Ω(n log(n))	Θ(n log(n))	O(n^2)	O(n)
Shell Sort (http://en.wikipedia.org/wiki/Shellsort)	Ω(n log(n))	θ(n(log(n))^2)	O(n(log(n))^2)	O(1)
Bucket Sort (http://en.wikipedia.org/wiki/Bucket_sort)	Ω(n+k)	Θ(n+k)	O(n^2)	O(n)
Radix Sort (http://en.wikipedia.org/wiki/Radix_sort)	Ω(nk)	Θ(nk)	O(nk)	O(n+k)
Counting Sort (https://en.wikipedia.org/wiki/Counting_sort)	Ω(n+k)	Θ(n+k)	O(n+k)	O(k)
Cubesort (https://en.wikipedia.org/wiki/Cubesort)	Ω(n)	Θ(n log(n))	O(n log(n))	O(n)

Recommend Reading

- How java manages Memory? (java-memory-model-and-garbage-collections.html)

- Why is it advised to use hashcode and equals together? (java-hashcode-and-equals.html)
- Comparable and Comparator in java (java-comparable-comparator.html)
- How to create Singleton class in Java? (java-singleton-class.html)
- Difference betwen equals and ==? (difference-between-equals-method-and-equality-operator-java.html)
- When to use abstract class over interface? (difference-between-abstract-class-and-interface-java.html)
- Difference between final, finally and finalize (difference-between-final-finally-and-finalize.html)
- Why is it recommended to ues Immutable objects in Java (java-immutable-class.html)

Subscribe to get latest updates.

Email Address *

Subscribe