DATA STRUCTURE	INSERTION	UPDATE	DELETION	SEARCH
Linked List	O(1) when it is on the head, and O(n) for the rest. At the beginning, update of pointers always happens.	O(n) - Insertion involves updating required traversing to specific node.	O(n) - Deletion involves updating pointers,traversing to specific node.	O(n) - Linear search through node.
Stack	O(1) - Insertion at the top.	No direct updating of elements in a stack.	Removing from the top.	O(n) - Linear search, not a primary use of stack.
Queue	O(1) - Insertion at the end	No direct updating of elements in a queue	Removing from the front	O(n) - Linear search, not a primary use of a queue
Hash table	O(1) average- Insertion involves hashing and placing in a bucket	O(1) average- Updating involves recomputing the hash and modifying the element	O(1) average- Deletion entails recomputing the hash and deleting the element	O(1) average- Search encompasses computing the hash and checking corresponding bucket
Sort	It depends on algorithm (e.g., quick sort's O(n log n) and merge sort for inserting and sorting elements.	Elements are generally not updated individually after sorting.		Depends on algorithm (e.g., O(log n) for binary search) - Efficient search in sorted data
Dictionary	O(1) average - Insertion involves hashing the key and placing in a bucket.	O(1) average - Updating involves recomputing the hash and modifying the value.	O(1) average- Deletion involves recomputing the hash and removing the key value pair	O(1) average - Searching involves computing the hash and checking the corresponding bucket.