Nilesh Domah domah001
Adeyinka Omotoyinbo omoto011

Method	**ArrayList Runtime**	**LinkedList Runtime**	**Explanation**
boolean add(T element)	O(n)	O(n)	AL: Iterates through the length of the array, n. LL: Checking for a linear condition in the while loop, n.
boolean add(int index, T element)	2O(n)	O(n)	AL: Iterates through the length of the array, n, twice (given both for loops). LL: Node increments as it iterates through the index in the while loop
void clear()	O(n)	O(1)	AL: Iterates through the length of the array, n. LL: Only clears the head, no iteration or recursion used
boolean contains(T element)	2O(n)	O(n)	AL: Multiple conditions for the first while loop as it checks from beginning to end (n). Then it iterates through the for loop for the length of the array, n. LL: Runs through the entire list, length n, looking for a non-null node in the while loop
T get(int index)	O(1)	O(n)	AL: Checks for a constant condition of index, which will only have that singular value. LL: Iterates through the list looking for a non null nude,

			list length of n, and checks for a specific condition
int indexOf(T element)	O(n)	O(n)	AL: Starting from beginning until hitting end, middle gets incremented. Also, under the else condition, the for loop iterates through the length of the array - 1, n. LL: same as T get(int index), goes thru list, checks for non null node
boolean isEmpty()	O(n)	O(1)	AL: Iterates through the length of the array, n. LL: checks constant condition of head of node
int lastIndexOf(T element)	O(n^2)	O(n)	AL: Starting from beginning until hitting end, middle increments in two different fashions, the second being within the first while loop. LL: Iterates through list of nodes checking if head is null
T set(int index, T element)	O(1)	O(n)	AL: Checks for constant condition with the if statement. LL: Checks constant condition of element and index, then iterates through list of nodes checking to see if current node is null
int size()	O(n)	O(n)	AL: Iterates through the length of the array, n, and checks if element is null. LL: Iterates through next node and sees if it is null

			(length n), then increments while doing that
void sort(boolean order)	O(n^2)	O(n^2)	AL: Iterates through the length of the array, n, to store element as stored. While doing that, checks conditions and iterates again for certain alphabetical order. LL: Iterates to check if next node is not null, while doing that it checks constant statements for an iterative condition
boolean remove(T element)	O(n^2)	O(n)	AL: Iterates through the length of the array, n, to check if i is null in a. Then it iterates through again and gets the next element. LL: Checks for two constant conditions of count size
T remove(int index)	O(n)	2O(n)	AL: Iterates through the length of the array, n. LL: Iterates twice to see if current node is not null