Method	ArrayList Runtime	LinkedList Runtime	Explanation		
boolean add(T element)	O(n)	O(1)	In the arrayList, theres a possibility that the array has to increase in size, resulting in traversing the original array. In the LinkedList, you can just add the element to the end of the list.		
boolean add(int index, T element)	O(2n)	O(n)	Both methods add an element somewhere in the middle of the list and have to shift indices to give room for the new element, however in the arrayList, there is a possibility that the array has to grow, resulting in O(2n)		
void clear()	O(1)	O(1)	Both methods have one line that clears their respective list.		
T get(int index)	O(1)	O(n)	You can access a element directly in a array, while in a linked list you have to traverse the list to access an element.		
int indexOf(T element)	O(n)	O(n)	Both lists traverse their respective lists to find the index of the element.		
boolean isEmpty()	O(1)	O(1)	Each method has one line that determines whether the list is empty or not.		
int size()	O(n)	O(n)	Both methods have to traverse their list and count how many elements there are.		
void sort()	O(n^2)	O(n^2)	Both methods use insertion sort to sort list. Therefore time complexities are the same.		
T remove(int index)	O(n)	O(n)	When an element is removed, both lists have to shift their indices to fill the empty gap.		
void greaterThan(T element)	O(n)	O(n)	Both methods have to traverse their list and remove any elements that are not greater than the given element.		
void lessThan(T element)	O(n)	O(n)	Both methods have to traverse their list and remove any elements that are not less than the given element.		
void equalTo(T element)	O(n)	O(n)	Both methods have to traverse their list and remove any elements that are not equal the given element.		
String toString()	O(n)	O(n)	Both methods access each element in their respective lists and displays each element on their own line.		
It shows that the LinkedList is more efficient than the ArrayList. This is mostly due that when an array is full, a new array has to be made that is larger in size and the original array values have to be copied to the new array.					