









Discussion:

Based on the graphed results, a linked list is more efficient than an array for deleting and inserting elements. On average there was a similar amount of reads for both data structures but, there were significantly less writes for the linked list in these cases. For both of these cases the array had to bump every element up or down an index causing a lot of reading and writing. The linked list had to search for the index which made the reads about equal with the array. Once the linked list reached the right index it only had to write once instead of bumping many elements.

Based on the graphed results, an array is more efficient than a linked list for replacing and swapping elements in the data structure. There was the same amount of writing but the array only had to read once and 3 times respectively where the linked list had to search for the index. The array the index location was known and could just reach the elements directly but the linked list's searching increased the amount of reading required. Once both of the data structures reached the element the writing was the same.