|  |  |  |
| --- | --- | --- |
| Dynamic Array | | |
| Elements | Time | Memory |
| 2^10 | 10 | 124 |
| 2^11 | 10 | 124 |
| 2^12 | 30 | 124 |
| 2^13 | 50 | 124 |
| 2^14 | 90 | 124 |
| 2^15 | 160 | 2172 |
| 2^16 | 340 | 2172 |
| 2^17 | 680 | 2172 |
| 2^18 | 1380 | 2308 |

|  |  |  |
| --- | --- | --- |
| Linked List | | |
| Elements | Time | Memory |
| 2^10 | 20ms | 1080kb |
| 2^11 | 80ms | 1080kb |
| 2^12 | 290ms | 1080kb |
| 2^13 | 1160ms | 1436kb |
| 2^14 | 4550ms | 2228kb |
| 2^15 | 18070ms | 4076kb |
| 2^16 | 71330ms | 10540kb |
| 2^17 | 293720ms | 20964kb |
| 2^18 |  | 41548kb |

1. The linked list uses more memory because it has to store the next and previous pointer for each link in the list.
2. The array is faster because it has a direct reference to every element in the array. The linked list has to navigate through each node with the internal pointers.
3. The Linked list would be faster with a remove function because it can just change the pointer of the previous and next links whereas the array would have to change the location for all of the elements in the array.