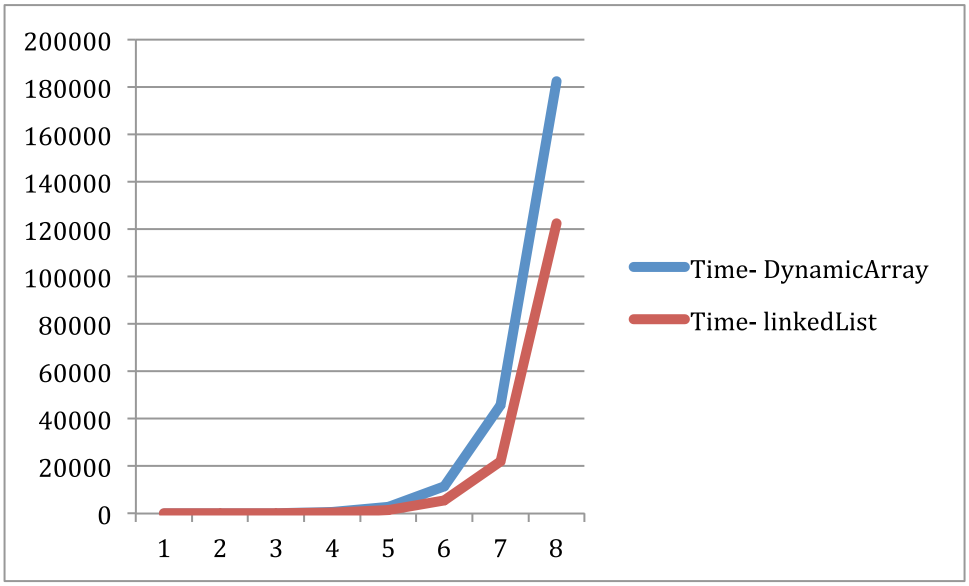
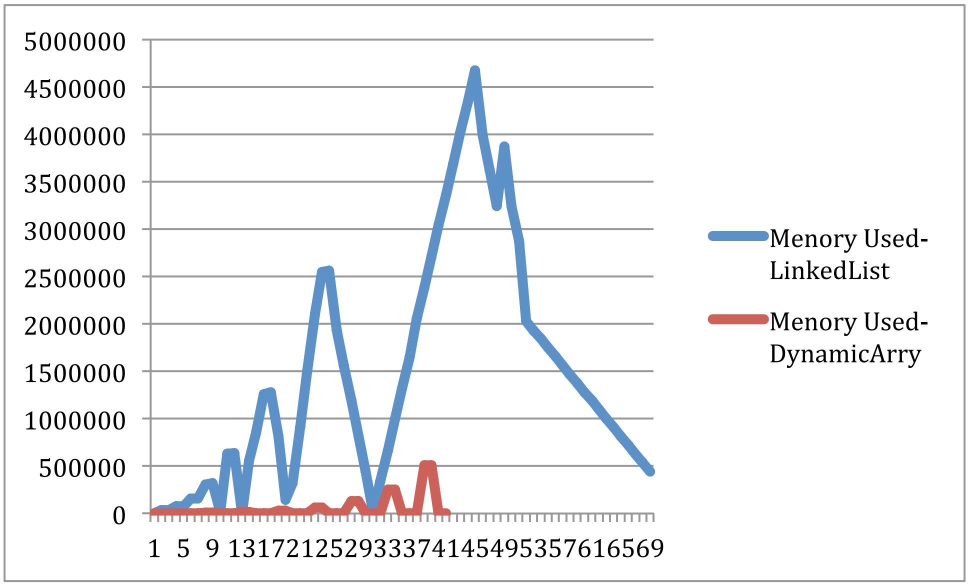


Elements	Time- DynamicArray	Time- linkedList
1000	10	10
2000	40	20
4000	180	80
8000	710	340
16000	2850	1340
32000	11420	5470
64000	45660	21970
128000	182560	122390



n	Menory Used-LinkedList	Menory Used-DynamicArry
0	0	0
1	39,640	24
2	40,120	4,032
3	79,400	4,032
4	80,120	24
5	157,760	0
6	160,120	24
7	310,280	8,032
8	320,120	8,032
9	15,320	24
10	629,720	0
11	640,120	24
12	1,000	16,032
13	556,080	16,032
14	853,040	24
15	1,261,360	0
16	1,280,120	24
17	816,840	32,032

18	143,240	32,032
19	319,280	24
20	913,200	0
21	1,507,120	24
22	2,101,040	64,032
23	2,546,480	64,032
24	2,560,120	24
25	1,936,280	0
26	1,565,080	24
27	1,193,880	128,032
28	822,680	128,032
29	451,480	24
30	11,160	0
31	338,520	24
32	665,880	256,032
33	993,240	256,032
34	1,320,600	24
35	1,647,960	0
36	2,057,160	24
37	2,384,520	512,032
38	2,711,880	512,032
39	3,039,240	24
40	3,366,600	0
41	3,693,960	
42	4,021,320	
43	4,348,680	
44	4,676,040	
45	3,996,880	
46	3,622,480	

47	3,248,080	
48	3,873,680	
49	3,248,080	
50	2,873,680	
51	2,031,280	
52	1,937,680	
53	1,844,080	
54	1,750,480	
55	1,656,880	
56	1,563,280	
57	1,469,680	
58	1,376,080	
59	1,282,480	
60	1,188,880	
61	1,095,280	
62	1,001,680	
63	908,080	
64	814,480	
65	720,880	
66	627,280	
67	533,680	
68	440,080	

1.Which of the implementations uses more memory? Explain why.

Answer: The Linked List uses more memory. The reason is that every linked list node need to create two extra pointer in order to point their previous node and next node. These two pointers also need to allocate the same memory as their linked list node. It means every time when we create a new node, there will be allocated three block of memory of linked list node. Regarding dynamic array, in the structure of dynamic array, there is only one pointer need to allocate. Additionally it just needs to allocate a chunk of new memory as it is filled. Therefore the dynamic array cost less memory than Linked list.

2. Which of the implementations is the fastest? Explain why.

Answer: The time cost of linked list and dynamic array is nearly same. The reason is that for the two data structure, performing contain() they all need to go through the whole data structure. Their big-oh complexity is all O(n).

3. Would you expect anything to change if the loop performed remove() instead of contains()? If so, what?

Answer: If performing the remove() instead of contains() in the loop, The linked list will faster than dynamic array. The reason is that when we perform remove() in linked list, for this test case, every time we can find the node that needs to be remove at front of linked list, so does dynamic array. But for dynamic array, when it removes an element from the front, it also need to shift other elements back one to fill the gap which means more cost.