Assignment 3

SYSC2100A
Professor Donald Bailey

Nathan MacDiarmid
101098993

Bubble Sort:

n	sorted list		reverse sorted list		random list	
	# compare	# swaps	# compare	# swaps	# compare	# swaps
10	45	0	45	45	45	23
100	4950	0	4950	4950	4950	2773
500	124750	0	124750	124750	124750	61370
1000	499500	0	499500	499500	499500	252109

Selection Sort:

n	sorted list		reverse sorted list		random list	
	# compare	# swaps	# compare	# swaps	# compare	# swaps
10	55	0	55	5	55	8
100	5050	0	5050	50	5050	96
500	120250	0	125250	250	125250	492
1000	500500	0	500500	500	500500	989

Merge Sort:

n	sorted list		reverse sorted list		random list	
	# compare	# swaps	# compare	# swaps	# compare	# swaps
10	15	34	19	34	21	34
100	316	672	356	672	539	672
500	2216	4488	2272	4488	3850	4488
1000	4932	9976	5044	9976	8712	9976

Heap Sort:

n	sorted list		reverse sorted list		random list	
	# compare	# swaps	# compare	# swaps	# compare	# swaps
10	41	30	35	21	42	28
100	1081	640	944	516	1023	579
500	7756	4354	7010	3676	7395	4013
1000	17582	9708	15965	8316	16858	9068

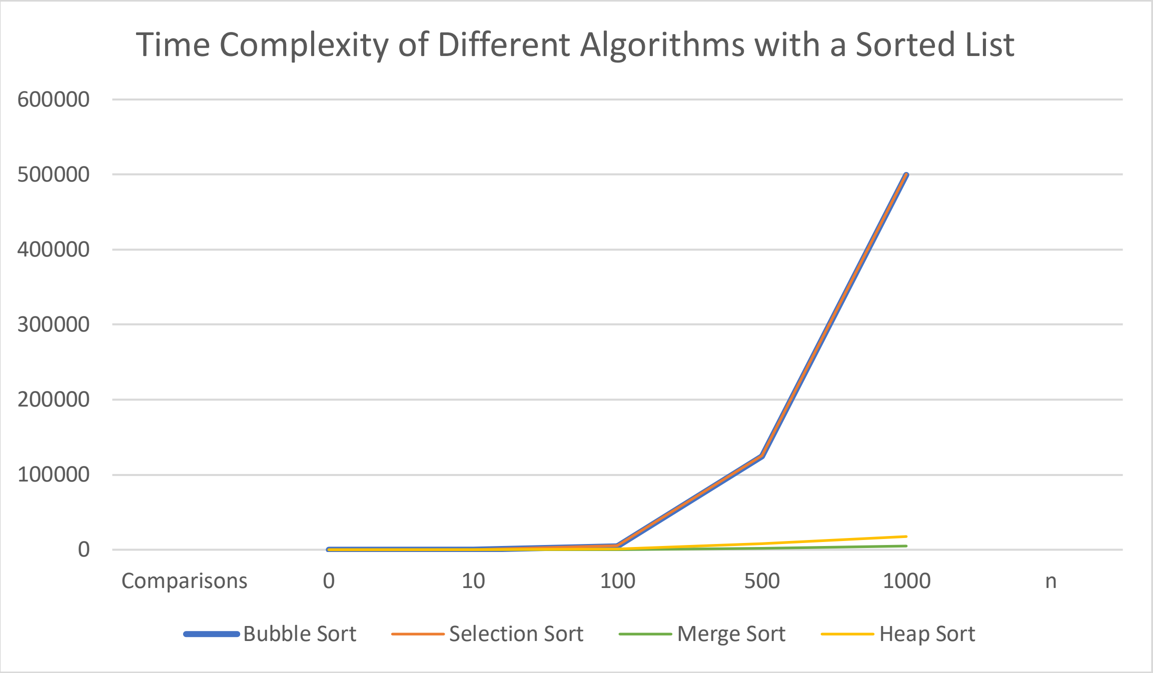

Table 1: Time Complexity of Different Algorithms with a Sorted List

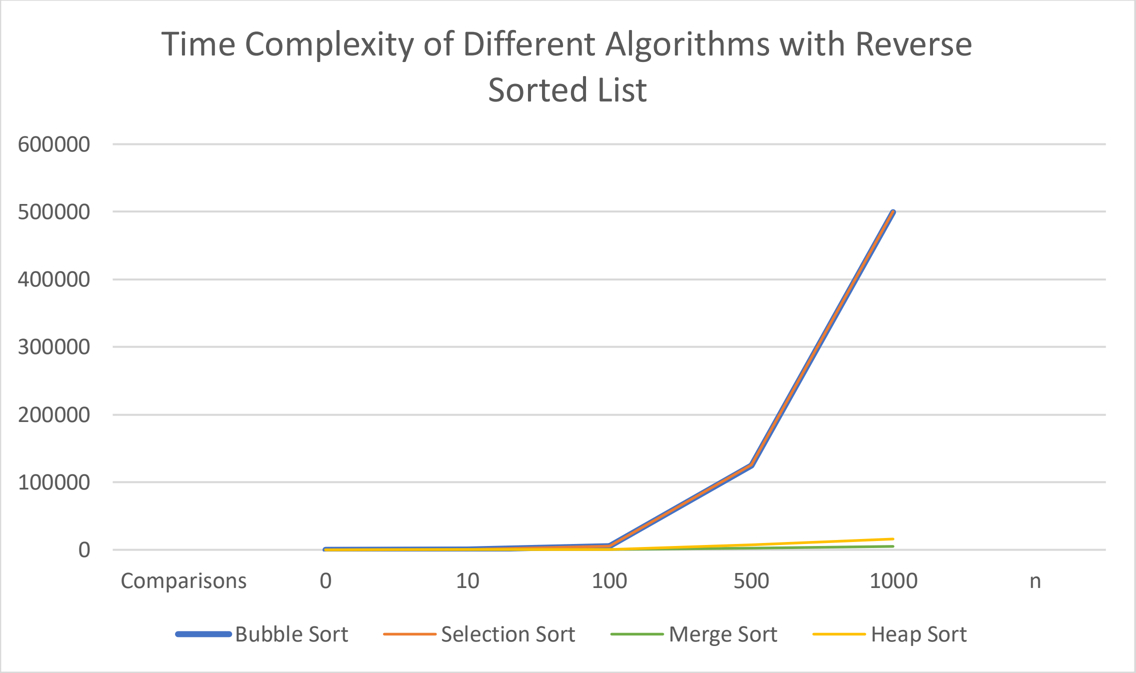

Table 2: Time Complexity of Different Alogrithms with Reverse Sorted List

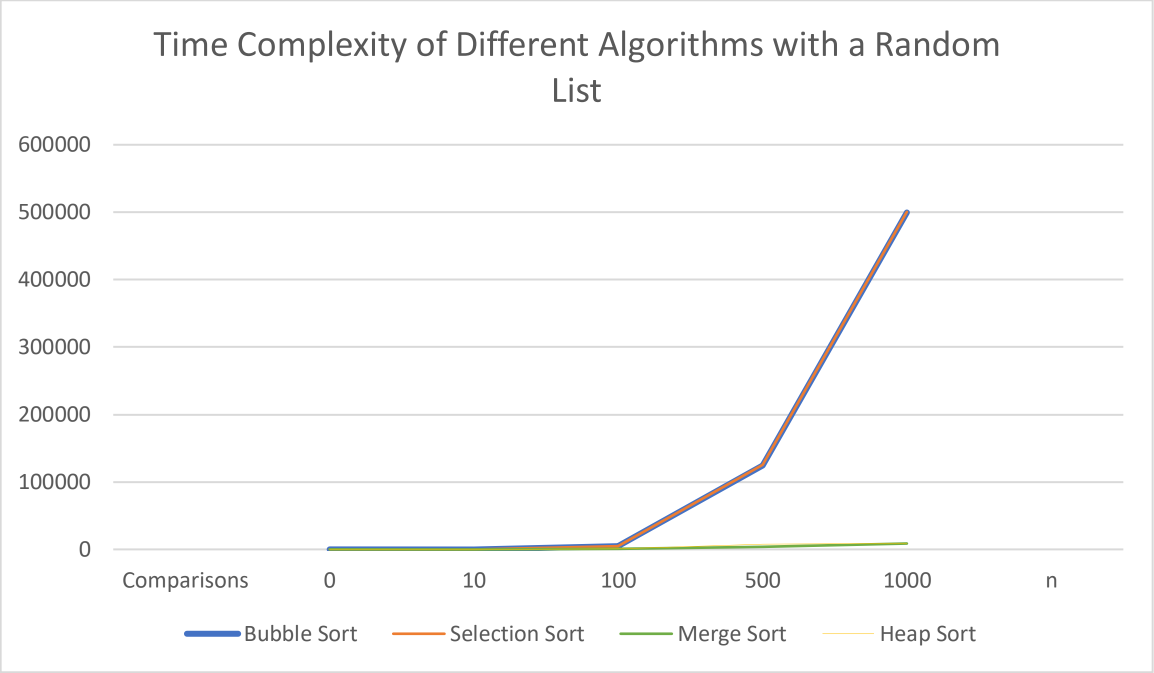

Table 3: Time Complexity of Different Algorithms with a Random List

Bubble Sort:

Bubble Sort Algorithm - BSA

Looking at the Bubble Sort Algorithm, it shows that when the number of items in a list get bigger, the more comparissons the BSA does. It is also observed that if the BSA is of O(n^2), the test cases for *n* may be inputed to this formula. In the reveresed ordered case, the result is: $500000 \approx \frac{1000^2}{2}$. Since O($\frac{n^2}{2}$) $\approx O(n^2)$ because of the definition given where $\frac{1}{2}$ is the constant. This proves the time complexity of the BSA to be O(n^2).

Selection Sort:

Selection Sort Algorithm – SSA

Looking at the Selection Sort Algorithm, it follows the same pattern and path as the BSA. For the reveresed ordered case, the result is: $500000 \approx \frac{1000^2}{2}$, which is the same as the BSA. Therefore, it can be concluded that the SSA is also O(n^2).

Merge Sort:

Merge Sort Algorithm – MSA

Looking at the Merge Sort Algorithm, it follows a different path than the BSA and SSA. It is also observed that if the MSA is of O(nlogn), the test cases for *n* may be inputed to this forumla. In the reveresed ordered case, the result is: 5000 ≈ $\frac{5}{3}$1000 log 1000. In this case, the constant is $\frac{5}{3}$ and proves that the MSA is of O(nlogn).

Heap Sort:

Heap Sort Algorithm – HSA

Looking at the Heap Sort Algorithm, it follows the same pattern as the MSA. For the reveresed ordered case, the result is: 17000 ≈ $\frac{17}{3}$1000 log 1000. In this case, the constant is $\frac{17}{3}$ which proves that the HSA is of O(nlogn).