Assignment 3 – Characterizing the Running Times of Sorting Algorithms

Wednesday, April 14, 2021

Prepared by Zakaria Ismail

Zakaria Ismail - 101143497

Assignment 3 – Characterizing the Running Times of Sorting Algorithms

1.0 PURPOSE

The purpose of this experiment is to verify the Big O characterizations of the bubble sort, heap sort, merge sort, and selection sort.

2.0 BACKGROUND

Big O notation gives the upper bound on the growth of a function. It is the worst-case performance of a function. In theory, the Big O characterization of:

- bubble sort is $O(n^2)$
- heap sort is $O(nlogn)$
- merge sort is $O(nlogn)$
- selection sort is $O(n^2)$

The complexity of an algorithm can be determined based on factors such as time or the number of operations performed, for example. In this experiment, the complexity of an algorithm will be determined by the upper bound of the sum of the number of comparisons and swaps. Only the largest degree term is taken into account in Big O notation.

3.0 EXPERIMENT

This experiment will use the Python programming language, alongside the matplotlib and pandas libraries in order to facilitate the management of the experimental data and the plotting.

The first part of the experiment will involve determining the worst-case scenario of each of bubble sort, heap sort, merge sort, and selection sort. The parameters to be measured are the number of comparisons and the number of swaps. The three scenarios to be compared will be when the input is a sorted (ascending) list, a reverse sorted (descending) list, and a random list. The same random number list will be used for each of the algorithms during each iteration. To do this, each scenario will be tested for each algorithm while incrementing the number of elements in the list. The worst-case scenario is defined as the case that has the largest upper bound number of comparisons plus swaps.

In this experiment, the maximum number of list elements will be 10 000 and the increment between iterations will be 200, starting from 0, giving 50 data points per plot.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
1		Num Items	t-Sorted-Co	sort-Sorted	-Reverse-S	ort-Reverse	-Random-	Sort-Random	Sorted-Co	sort-Sorted	-Reverse-C	ort-Reverse	Random-C	ort-Random	t-Sorted-Co	sort-Sorted	-Reverse-C	ort-Reverse	-Random-C	ort-Random	rt-Sorted-C	sort-Sorted	t-Reverse-	sort-Reverse	t-Random-(	sort-Random	-Swaps
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	1	200	19900	0	19900	19900	19900	9558	2587	1289	2275	1007	2445	1155	732	0	812	812	1277	649	20100	0	20100	100	20100	194	
4	2	400	79800	0	79800	79800	79800	39642	5984	2973	5352	2423	5694	2711	1664	0	1824	1824	2951	1510	80200	0	80200	200	80200	394	
5	3	600	179700	0	179700	179700	179700	85601	9693	4805	8718	3973	9284	4448	2660	0	2916	2916	4774	2443	180300	0	180300	300	180300	594	
6	4	800	319600	0	319600	319600	319600	165122	13639	6755	12293	5643	13002	6215	3728	0	4048	4048	6711	3426	320400	0	320400	400	320400	790	
7	5	1000	499500	0	499500	499500	499500	241073	17583	8709	15965	7317	16844	8072	4932	0	5044	5044	8736	4397	500500	0	500500	500	500500	993	
8	6	1200	719400	0	719400	719400	719400	357378	21856	10805	19823	9141	20905	10010	5920	0	6432	6432	10788	5526	720600	0	720600	600	720600	1185	
9	7	1400	979300	0	979300	979300	979300	491611	26240	12965	23765	11005	24995	11992	7044	0	7708	7708	12853	6598	980700	0	980700	700	980700	1390	
10	8	1600	1279200	0	1279200	1279200	1279200	675218	30516	15063	27750	12873	29181	14042	8256	0	8896	8896	15059	7673	1280800	0	1280800	800	1280800	1591	
11	9	1800	1619100	0	1619100	1619100	1619100	823418	34928	17269	31904	14809	33447	16080	9508	0	10044	10044	17209	8690	1620900	0	1620900	900	1620900	1795	
12	10	2000	1999000	0	1999000	1999000	1999000	990025	39159	19301	35964	16709	37731	18182	10864	0	11088	11088	19364	9704	2001000	0	2001000	1000	2001000	1987	
13	11	2200	2418900	0	2418900	2418900	2418900	1202884	43924	21745	40215	18689	42144	20859	11940	0	12564	12564	21685	11042	2421100	0	2421100	1100	2421100	2195	
14	12	2400	2878800	0	2878800	2878800	2878800	1420132	48666	24119	44541	20701	46624	22470	13040	0	14064	14064	23976	12265	2881200	0	2881200	1200	2881200	2392	
15	13	2600	3378700	0	3378700	3378700	3378700	1732690	53403	26505	48835	22685	51138	24675	14260	0	15444	15444	26243	13426	3381300	0	3381300	1300	3381300	2588	
16	14	2800	3918600	0	3918600	3918600	3918600	1933427	58147	28873	53222	24731	55653	26881	15488	0	16816	16816	28614	14633	3921400	0	3921400	1400	3921400	2793	
17	15	3000	4498500	0	4498500	4498500	4498500	2245906	62875	31253	57537	26751	60212	29061	16828	0	18076	18076	30946	15702	4501500	0	4501500	1500	4501500	2996	
18	16	3200	5118400	0	5118400	5118400	5118400	2555767	67615	33629	62072	28917	64854	31343	18112	0	19392	19392	33252	16919	5121600	0	5121600	1600	5121600	3191	
19	17	3400	5778300	0	5778300	5778300	5778300	2871466	72362	36011	66538	30931	69379	33470	19372	0	20732	20732	35715	18217	5781700	0	5781700	1700	5781700	3393	
20	18	3600	6478200	0	6478200	6478200	6478200	3201390	77095	38389	71115	33147	74095	35815	20816	0	21888	21888	38100	19370	6481800	0	6481800	1800	6481800	3594	
21	19	3800	7218100	0	7218100	7218100	7218100	3569852	81832	40759	75641	35279	78792	38096	22188	0	23116	23116	40377	20467	7221900	0	7221900	1900	7221900	3790	
22	20	4000	7998000	0	7998000	7998000	7998000	3959272	86576	43143	80151	37437	83363	40265	23728	0	24176	24176	42811	21514	8002000	0	8002000	2000	8002000	3994	
23	21	4200	8817900	0	8817900	8817900	8817900	4465734	91530	45615	84681	39553	88070	42543	25012	0	25596	25596	45317	22876	8822100	0	8822100	2100	8822100	4193	
24	22	4400	9677800	0	9677800	9677800	9677800	4860854	96687	48135	89347	41799	93007	44976	26080	0	27328	27328	47674	24220	9682200	0	9682200	2200	9682200	4390	
25	23	4600	10577700	0	10577700	10577700	10577700	5252225	101818	50773	94018	44053	97915	47406	27324	0	28884	28884	50188	25469	10582300	0	10582300	2300	10582300	4595	
26	24	4800	11517600	0	11517600	11517600	11517600	5616117	106975	53347	98672	46219	102762	49732	28480	0	30528	30528	52746	26875	11522400	0	11522400	2400	11522400	4792	
27	25	5000	12497500	0	12497500	12497500	12497500	6193633	112126	55933	103227	48437	107628	52086	29804	0	32004	32004	55195	28130	12502500	0	12502500	2500	12502500	4989	
28	26	5200	13517400	0	13517400	13517400	13517400	6723328	117279	58511	107990	50667	112667	54586	31120	0	33488	33488	57700	29474	13522600	0	13522600	2600	13522600	5192	
29	27	5400	14577300	0	14577300	14577300	14577300	7252272	122442	61105	112753	52921	117639	56927	32380	0	35028	35028	60230	30746	14582700	0	14582700	2700	14582700	5386	
30	28	5600	15677200	0	15677200	15677200	15677200	7795649	127587	63665	117440	55151	122422	59236	33776	0	36432	36432	62846	32046	15682800	0	15682800	2800	15682800	5595	
31	29	5800	16817100	0	16817100	16817100	16817100	8463010	132728	66253	122168	57337	127523	61693	35140	0	37868	37868	65335	33320	16822900	0	16822900	2900	16822900	5789	
32	30	6000	17997000	0	17997000	17997000	17997000	8930554	137876	68829	126920	59599	132551	64241	36656	0	39152	39152	67857	34400	18003000	0	18003000	3000	18003000	5986	
33	31	6200	19216900	0	19216900	19216900	19216900	9692004	143026	71417	131660	61849	137378	66491	38156	0	40452	40452	70465	35611	19223100	0	19223100	3100	19223100	6190	
34	32	6400	20476800	0	20476800	20476800	20476800	10181212	148187	74005	136762	64359	142509	68990	39424	0	41984	41984	72910	37079	20483200	0	20483200	3200	20483200	6390	
35	33	6600	21776700	0	21776700	21776700	21776700	10801984	153323	76575	141578	66531	147457	71418	40756	0	43452	43452	75556	38393	21783300	0	21783300	3300	21783300	6592	
36	34	6800	23116600	0	23116600	23116600	23116600	11433250	158473	79149	146519	68865	152495	73899	42144	0	44864	44864	78059	39697	23123400	0	23123400	3400	23123400	6789	
37	35	7000	24496500	0	24496500	24496500	24496500	12363348	163651	81765	151408	71171	157626	76331	43628	0	46180	46180	80698	41046	24503500	0	24503500	3500	24503500	6987	
38	36	7200	25916400	0	25916400	25916400	25916400	12744993	168778	84311	156472	73575	162609	78856	45232	0	47376	47376	83236	42113	25923600	0	25923600	3600	25923600	7187	
39	37	7400	27376300	0	27376300	27376300	27376300	13517769	173924	86897	161239	75801	167595	81211	46636	0	48772	48772	85786	43423	27383700	0	27383700	3700	27383700	7390	
40	38	7600	28876200	0	28876200	28876200	28876200	14506010	179073	89471	166155	78113	172744	83759	48176	0	50032	50032	88436	44621	28883800	0	28883800	3800	28883800	7584	
41	39	7800	30416100	0	30416100	30416100	30416100	15334931	184224	92055	171012	80417	177792	86221	49780	0	51228	51228	91138	46072	30423900	0	30423900	3900	30423900	7789	
42	40	8000	31996000	0	31996000	31996000	31996000	16043740	189377	94639	176057	82777	182711	88567	51456	0	52352	52352	93630	47038	32004000	0	32004000	4000	32004000	7987	
43	41	8200	33615900	0	33615900	33615900	33615900	16906042	194535	97221	180849	85057	188009	91210	53268	0	53348	53348	96307	48241	33624100	0	33624100	4100	33624100	8192	
44	42	8400	35275800	0	35275800	35275800	35275800	17369390	200068	99963	185905	87381	193202	93786	54224	0	55392	55392	98950	49879	35284200	0	35284200	4200	35284200	8386	
45	43	8600	36975700	0	36975700	36975700	36975700	18457121	205630	102713	191087	89919	198407	96131	55356	0	57260	57260	101706	51512	36984300	0	36984300	4300	36984300	8584	
46	44	8800	38715600	0	38715600	38715600	38715600	19216460	211160	105451	196208	92279	203491	98703	56560	0	59056	59056	104302	52857	38724400	0	38724400	4400	38724400	8791	
47	45	9000	40495500	0	40495500	40495500	40495500	19918942	216720	108223	201200	94687	208921	101274	57820	0	60796	60796	107062	54361	40504500	0	40504500	4500	40504500	8990	
48	46	9200	42315400	0	42315400	42315400	42315400	21255890	222269	110969	206395	97099	214187	103829	59248	0	62368	62368	109619	55715	42324600	0	42324600	4600	42324600	9187	
49	47	9400	44175300	0	44175300	44175300	44175300	22106832	227810	113703	211398	99571	219391	106344	60420	0	64196	64196	112389	57218	44184700	0	44184700	4700	44184700	9392	
50	48	9600	46075200	0	46075200	46075200	46075200	23198860	233358	116451	216601	101947	224922	109177	61760	0	65856	65856	114971	58363	46084800	0	46084800	4800	46084800	9591	
51	49	9800	48015100	0	48015100	48015100	48015100	24130773	238909	119191	221656	104395	229967	111511	63348	0	67468	67468	117751	60102	48024900	0	48024900	4900	48024900	9790	
52	50	10000	49995000	0	49995000	49995000	49995000	25075577	244460	121957	226682	106697	235263	114155	64608	0	69008	69008	120465	61398	50005000	0	50005000	5000	50005000	9995	

Figure 1. Experimental data

3.1 Bubble Sort

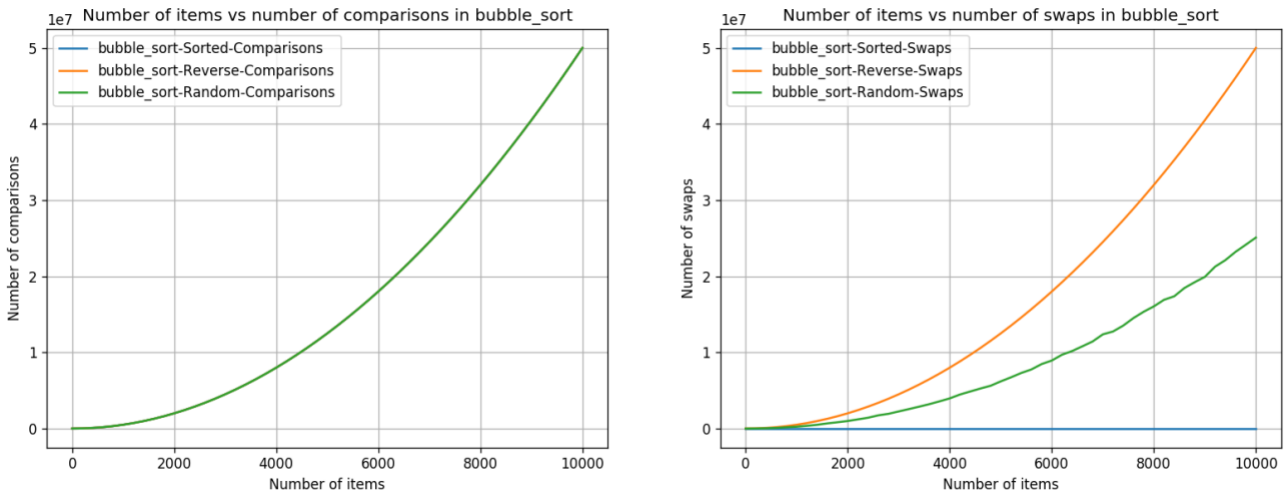

Figure 2. Plots for the number of comparisons and swaps against the number of items in bubble sort

The number of comparisons in bubble sort for a sorted, reverse, and random list are equal no matter the number of items in the list and appear to be quadratic. The number of swaps in bubble sort is at its greatest when the algorithm is performed on a reverse list and appears to be quadratic. Therefore, the worst-case scenario for the bubble sort algorithm is a reverse list. The total number of computations in bubble sort can be expressed as a function of f:

$$f(n) = \#comparisons + \#swaps$$

$$= a * n^2 + b * n^2$$

$$= (a + b) * n^2$$

$$= c * n^2$$

$$f(n) \leq O(n^2)$$

Where a, b, and c are arbitrary constants. Therefore, the complexity of bubble sort appears to be $O(n^2)$.

3.2 Heap Sort

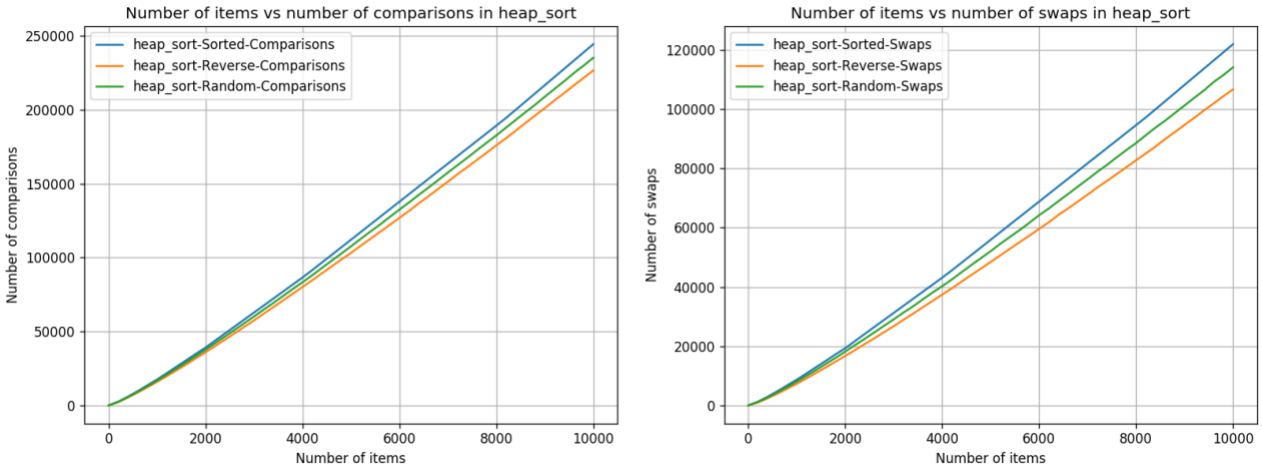

Figure 3. Plots for the number of comparisons and the number of swaps against the number of items in heap sort

The number of comparisons in heap sort is at its greatest when the list is sorted, which appears to be log-linear. This is also true for the number of swaps, which also appear log-linear.

Therefore, the worst-case scenario for heap sort algorithm is when the list is sorted.

The total number of computations can be expressed as a function of f:

$$f(n) = \#comparisons + \#swaps$$

$$= a * nlogn + b * nlogn$$

$$= (a + b) * nlogn$$

$$= c * nlogn$$

$$f(n) \leq O(nlogn)$$

Where a, b and c, are arbitrary constants. Therefore, the complexity of heap sort appears to be $O(nlogn)$.

3.3 Merge Sort

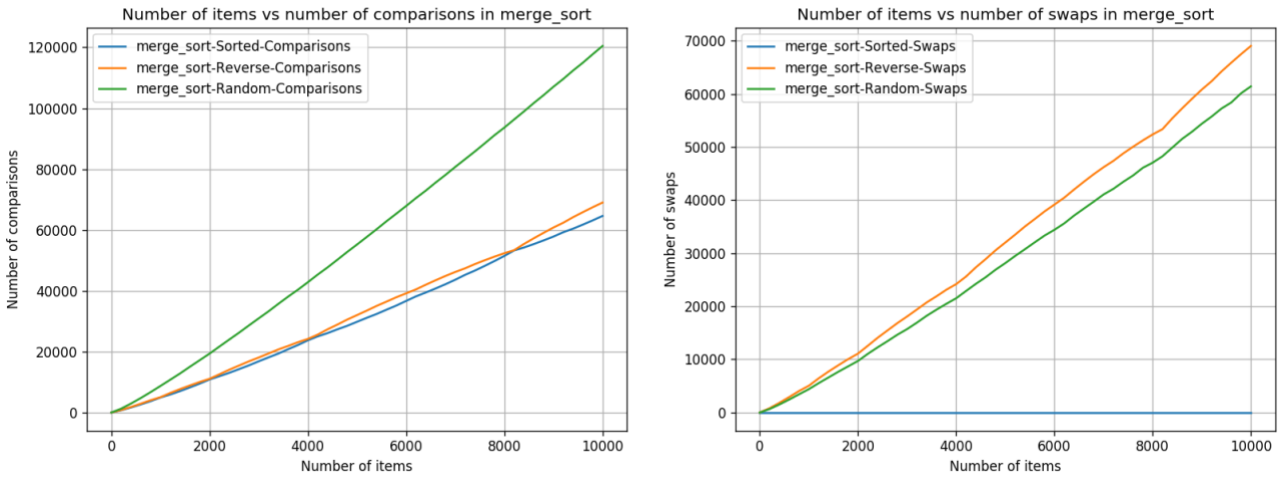

Figure 4. Plots for the number of comparisons and the number of swaps against the number of items in merge sort

The number of comparisons in merge sort is at its greatest when the list is random, which appears log-linear. The number of swaps is at its greatest when the list is reversed, which appears log-linear. Since the number of comparisons is more dominant than the number of swaps for every value of n, the worst-case scenario for the merge sort algorithm is when the list is random.

The total number of computations can be expressed as a function of f:

$$f(n) = \#comparisons + \#swaps$$

$$= a * nlogn + b * nlogn$$

$$= (a + b) * nlogn$$

$$= c * nlogn$$

$$f(n) \leq O(nlogn)$$

Where a, b, and c are arbitrary constants. Therefore, the complexity of merge sort appears to be $O(nlogn)$.

3.4 Selection Sort

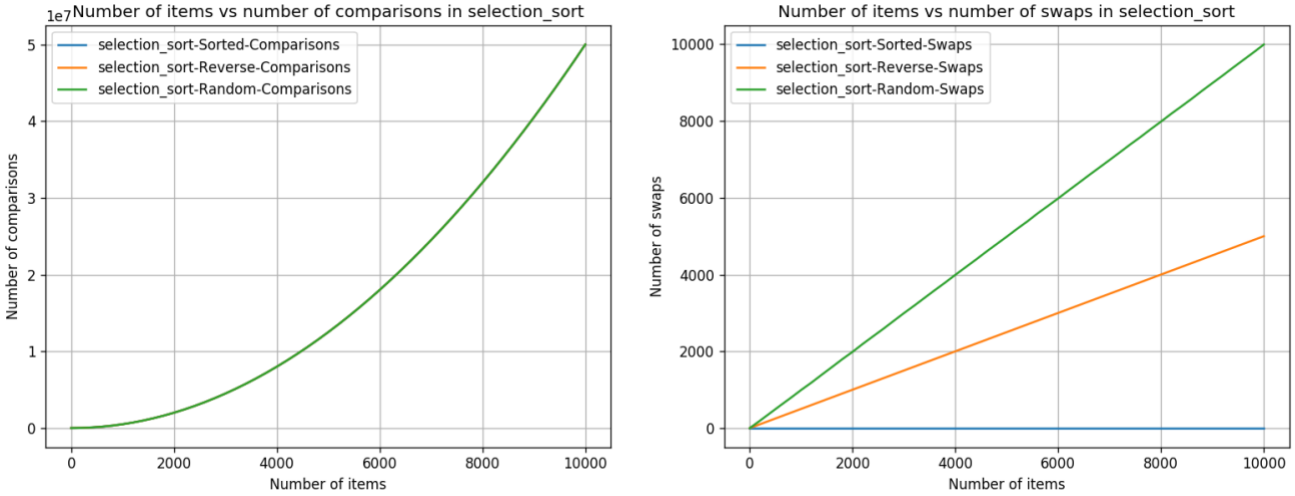

Figure 5. Plots for the number of comparisons and the number of swaps against the number of items in selection sort

The number of comparisons in bubble sort for a sorted, reverse, and random list are equal no matter the number of items in the list, which appear quadratic. The number of swaps in bubble sort is at its greatest when the algorithm is performed on a random list, which appears linear. Therefore, the worst-case scenario for the bubble sort algorithm is a random list.

The total number of computations can be expressed as a function of f:

$$f(n) = \#comparisons + \#swaps$$

$$= a * n^2 + b * n$$

$$\leq a * n^2 + d * n^2$$

$$\leq (a + d) * n^2$$

$$\leq c * n^2$$

$$f(n) \leq O(n^2)$$

Where a, b, c, and d, are arbitrary constants. Therefore, the complexity of bubble sort appears to be $O(n^2)$.

4.0 CONCLUSION

To conclude, the experimental results succeeded in verifying that:

- bubble sort is $O(n^2)$
- heap sort is $O(nlogn)$
- merge sort is $O(nlogn)$
- selection sort is $O(n^2)$

Contrary to intuition, the worst-case scenario for every algorithm was not the case when the input was a reverse (descending) list. For example, heap sort's upper bound case was when the input list was sorted. Furthermore, it is observed that there is a drastic difference in the number computations in a $O(n^2)$ algorithm compared to a $O(nlogn)$, which justifies the difference in time needed to process the same input, especially at very large values of n. For example, at n = 10 000, bubble sort's worst-case number of comparisons is approximately $5 * 10^7$, whereas merge sort's worst-case number of comparisons is approximately $2.5 * 10^5$; bubble sort requires approximately 200x more comparisons than merge sort when n = 10 000.