Ramiz Faragalla

Sorted Input

n	Quick Sort (Run Time | Num steps)		Randomized Quick Sort (Run Time | Num steps)		Heap Sort (Run Time | Num steps)		Insertion Sort (Run Time | Num steps)		Merge Sort (Run Time | Num Steps)	
100	360 microseconds	41185 steps	64 microseconds	4798 steps	90 microseconds	7650 steps	2 microseconds	596 steps	64 microseconds	7676 steps
200	1321 microseconds	162385 steps	115 microseconds	11008 steps	347 microseconds	17797 steps	2 microseconds	1196 steps	63 microseconds	17164 steps
300	3190 microseconds	363585 steps	377 microseconds	20118 steps	349 microseconds	28818 steps	3 microseconds	1796 steps	94 microseconds	27308 steps
400	5220 microseconds	644785 steps	251 microseconds	27006 steps	620 microseconds	40387 steps	3 microseconds	2396 steps	79 microseconds	37940 steps
500	5141 microseconds	1005985 steps	237 microseconds	31180 steps	380 microseconds	52147 steps	4 microseconds	2996 steps	301 microseconds	48436 steps
1000	25001 microseconds	4011985 steps	681 microseconds	71028 steps	823 microseconds	116446 steps	6 microseconds	5996 steps	250 microseconds	105884 steps
4000	331484 microseconds	64047985 steps	1982 microseconds	347666 steps	3937 microseconds	564465 steps	81 microseconds	23996 steps	889 microseconds	495572 steps
10000	1916020 microseconds	400119985 steps	4656 microseconds	995648 steps	10724 microseconds	1579911 steps	58 microseconds	59996 steps	2195 microseconds	1366932 steps

Reversely Sorted Input

n	Quick Sort (Run Time | Num steps)		Randomized Quick Sort (Run Time | Num steps)		Heap Sort (Run Time | Num steps)		Insertion Sort (Run Time | Num steps)		Merge Sort (Run Time | Num Steps)	
100	181 microseconds	28685 steps	67 microseconds	4678 steps	76 microseconds	6439 steps	24 microseconds	20396 steps	34 microseconds	7596 steps
200	796 microseconds	112385 steps	196 microseconds	10622 steps	284 microseconds	15049 steps	97 microseconds	80796 steps	241 microseconds	17004 steps
300	1588 microseconds	251085 steps	189 microseconds	18320 steps	371 microseconds	24723 steps	203 microseconds	181196 steps	72 microseconds	27052 steps
400	3345 microseconds	444785 steps	256 microseconds	24937 steps	393 microseconds	34949 steps	268 microseconds	321596 steps	79 microseconds	37620 steps
500	2781 microseconds	693485 steps	217 microseconds	34775 steps	325 microseconds	45527 steps	414 microseconds	501996 steps	128 microseconds	48324 steps
1000	18661 microseconds	2761985 steps	704 microseconds	70381 steps	719 microseconds	102644 steps	1626 microseconds	2003996 steps	240 microseconds	105660 steps
4000	163825 microseconds	44047985 steps	1860 microseconds	353431 steps	3559 microseconds	508453 steps	26100 microseconds	32015996 steps	997 microseconds	494676 steps
10000	1001087 microseconds	275119985 steps	4755 microseconds	967250 steps	10116 microseconds	1428607 steps	151165 microseconds	200039996 steps	2188 microseconds	1358132 steps

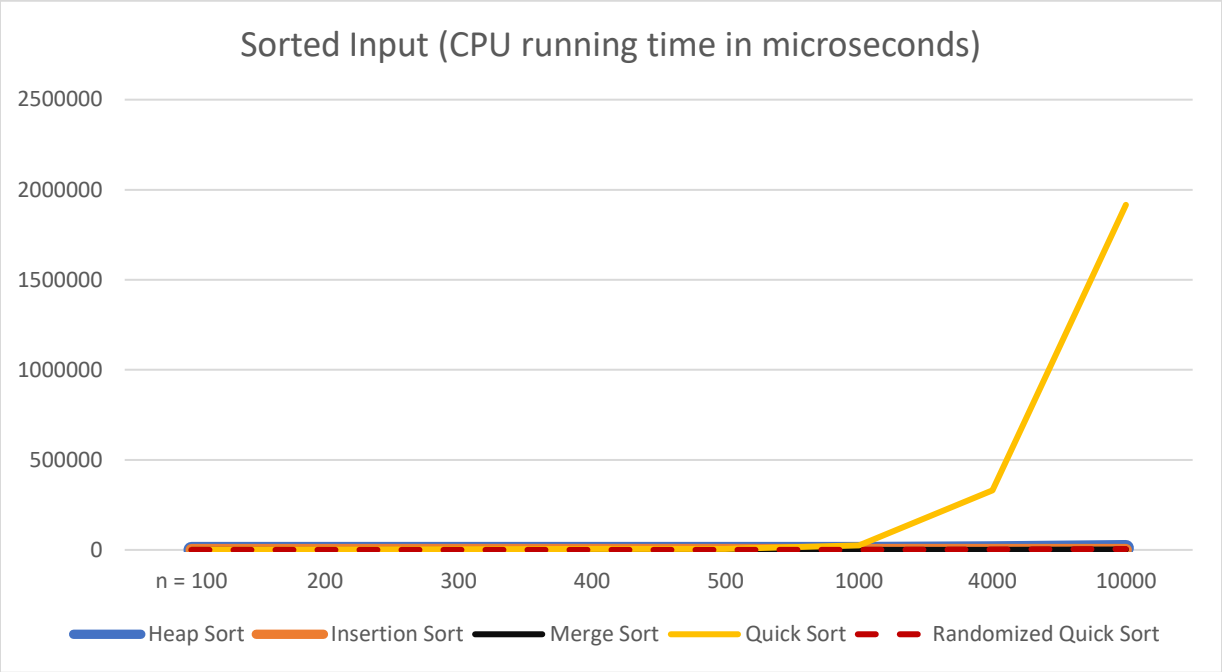
Ramiz Faragalla

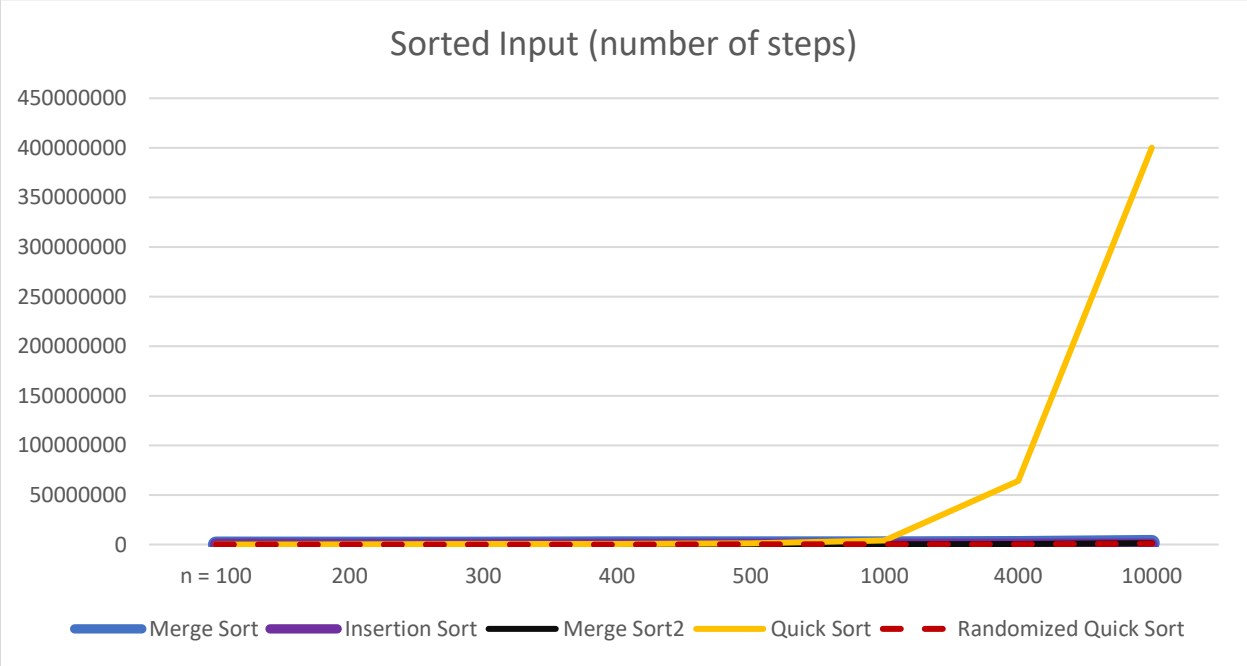
Random Permutation

n	Quick Sort (Run Time | Num steps)		Randomized Quick Sort (Run Time | Num steps)		Heap Sort (Run Time | Num steps)		Insertion Sort (Run Time | Num steps)		Merge Sort (Run Time | Num Steps)	
100	45 microseconds	4687 steps	157 microseconds	5035 steps	88 microseconds	7143 steps	21 microseconds	12996 steps	27 microseconds	8042 steps
200	237 microseconds	9734 steps	287 microseconds	11833 steps	201 microseconds	16488 steps	61 microseconds	51748 steps	73 microseconds	18024 steps
300	150 microseconds	18893 steps	205 microseconds	18662 steps	317 microseconds	26788 steps	108 microseconds	109304 steps	78 microseconds	28812 steps
400	186 microseconds	23823 steps	279 microseconds	26537 steps	403 microseconds	37979 steps	192 microseconds	188952 steps	100 microseconds	40182 steps
500	163 microseconds	29050 steps	222 microseconds	33082 steps	356 microseconds	49126 steps	242 microseconds	292044 steps	126 microseconds	51592 steps
1000	519 microseconds	66926 steps	718 microseconds	76708 steps	1093 microseconds	110744 steps	946 microseconds	1118244 steps	287 microseconds	112958 steps
4000	1803 microseconds	336564 steps	2264 microseconds	375210 steps	3987 microseconds	536400 steps	11078 microseconds	13013756 steps	986 microseconds	529700 steps
10000	4791 microseconds	946081 steps	5037 microseconds	926752 steps	10348 microseconds	1503151 steps	64236 microseconds	84415316 steps	2329 microseconds	1444832 steps

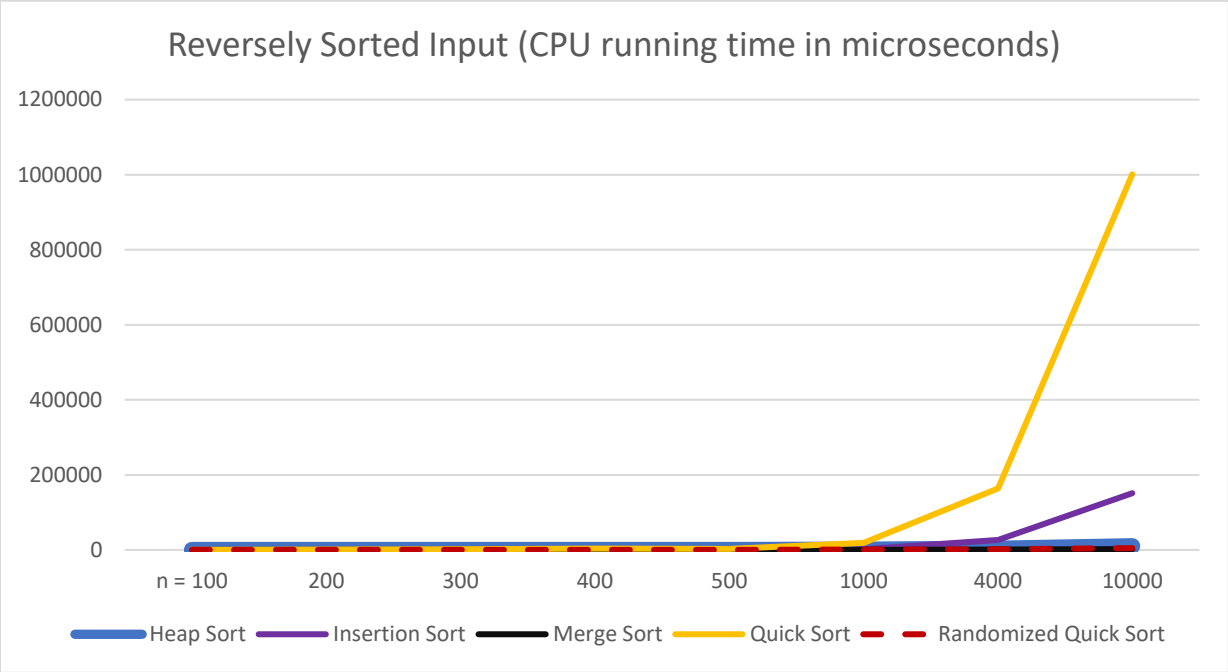
50 instances of n random numbers generated in the range of [1..n]

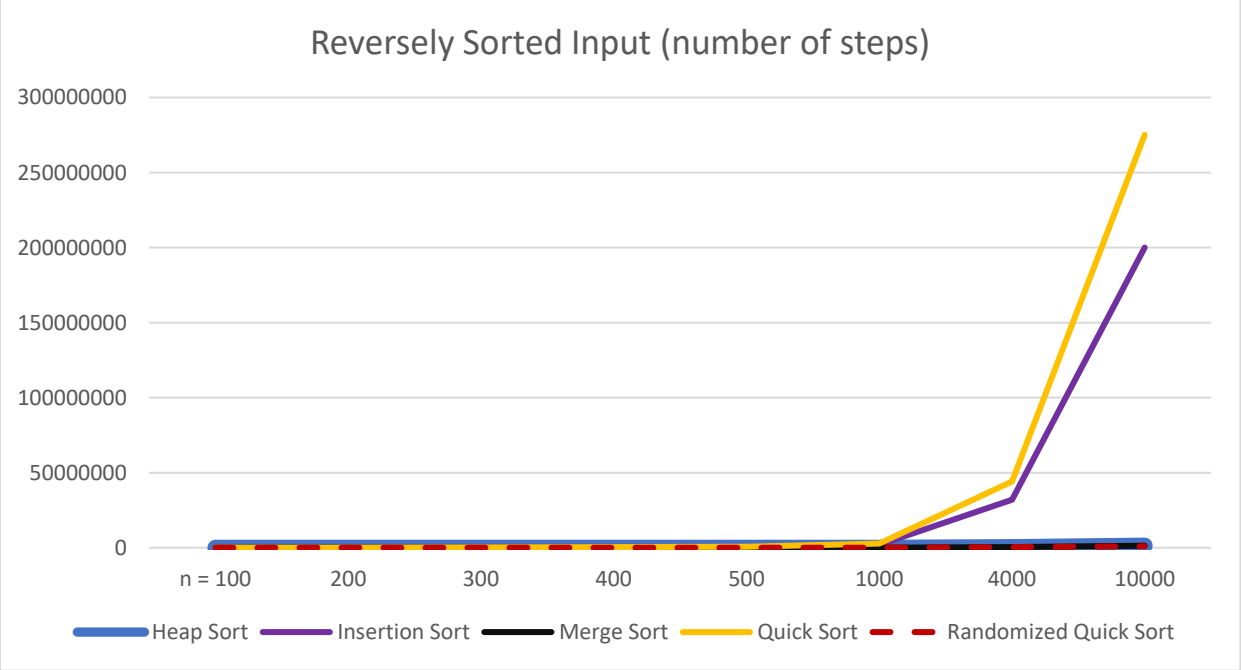
n	Quick Sort (Run Time | Num steps)		Randomized Quick Sort (Run Time | Num steps)		Heap Sort (Run Time | Num steps)		Insertion Sort (Run Time | Num steps)		Merge Sort (Run Time | Num Steps)	
100	40.92 microseconds	4511.48 steps	58.4 microseconds	4906.26 steps	116.22 microseconds	7084.32 steps	12.5 microseconds	10328 steps	25.12 microseconds	8050 steps
200	86.94 microseconds	10694.5 steps	127.52 microseconds	11290.6 steps	200.1 microseconds	16542.8 steps	44.58 microseconds	37092 steps	61.3 microseconds	18078 steps
300	149.92 microseconds	17265.9 steps	191.96 microseconds	18176.2 steps	356.4 microseconds	26874 steps	79.72 microseconds	94496 steps	64.1 microseconds	28868 steps
400	141.52 microseconds	23874.3 steps	184.24 microseconds	25302.3 steps	331 microseconds	37853.9 steps	139.26 microseconds	156344 steps	97.6 microseconds	40250 steps
500	170.94 microseconds	31130.5 steps	225.1 microseconds	32850.4 steps	396.24 microseconds	49145.5 steps	222.16 microseconds	243448 steps	122.06 microseconds	51594 steps
1000	385.16 microseconds	69187 steps	502.64 microseconds	73882.4 steps	856.18 microseconds	110164 steps	826.42 microseconds	968080 steps	260.18 microseconds	113252 steps
4000	1697.96 microseconds	340737 steps	2121.98 microseconds	355292 steps	3734.86 microseconds	536602 steps	12837.5 microseconds	1.60323e+07 steps	1119.68 microseconds	532932 steps
10000	4548.1 microseconds	955808 steps	5505.5 microseconds	991470 steps	10418.6 microseconds	1.50271e+06 steps	77212.5 microseconds	1.40149e+07 steps	2828.26 microseconds	1.46991e+06 steps


Sorted Input (CPU running time in microseconds)
2500000
2000000
1500000
1000000
500000
0
n = 100
200
300
400
500
1000
4000
10000
Heap Sort
Insertion Sort
Merge Sort
Quick Sort
Randomized Quick Sort

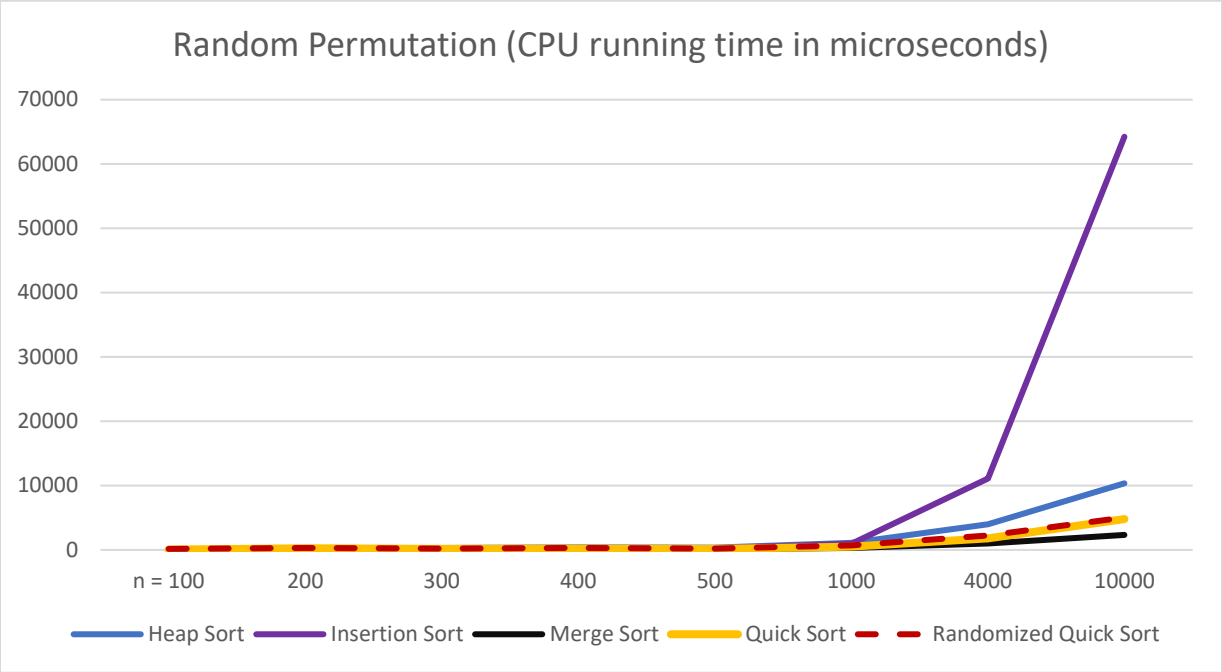

Sorted Input (number of steps)
450000000
400000000
350000000
300000000
250000000
200000000
150000000
100000000
50000000
0
n = 100
200
300
400
500
1000
4000
10000
Merge Sort
Insertion Sort
Merge Sort2
Quick Sort
Randomized Quick Sort

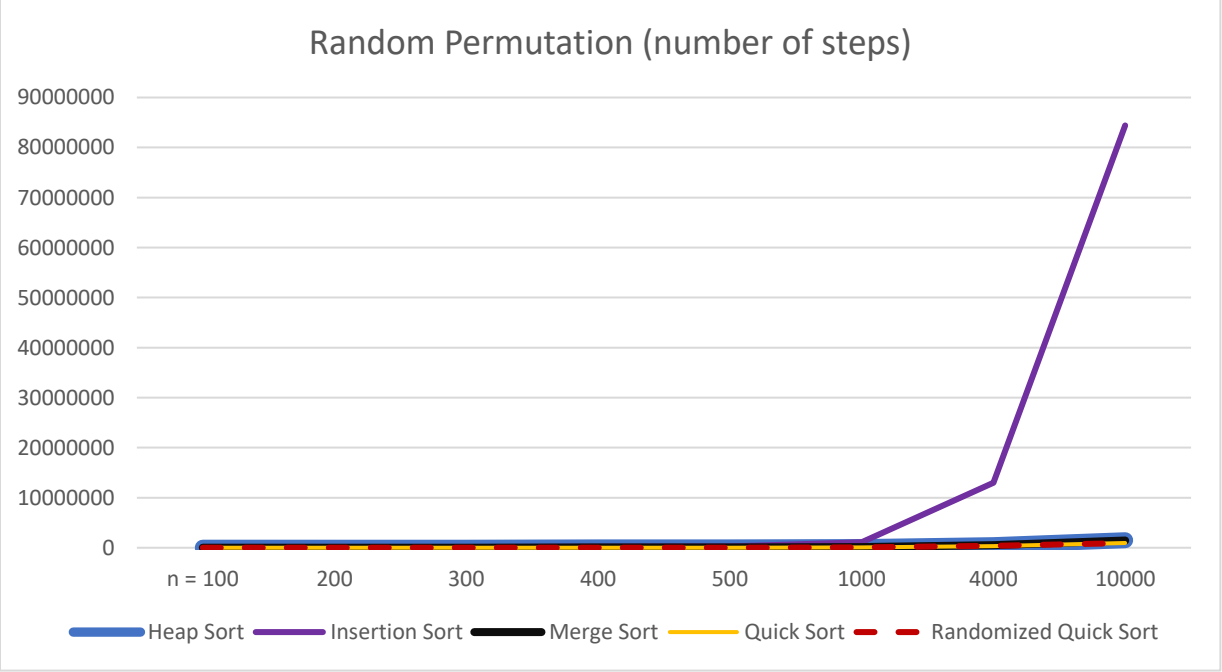
Ramiz Faragalla

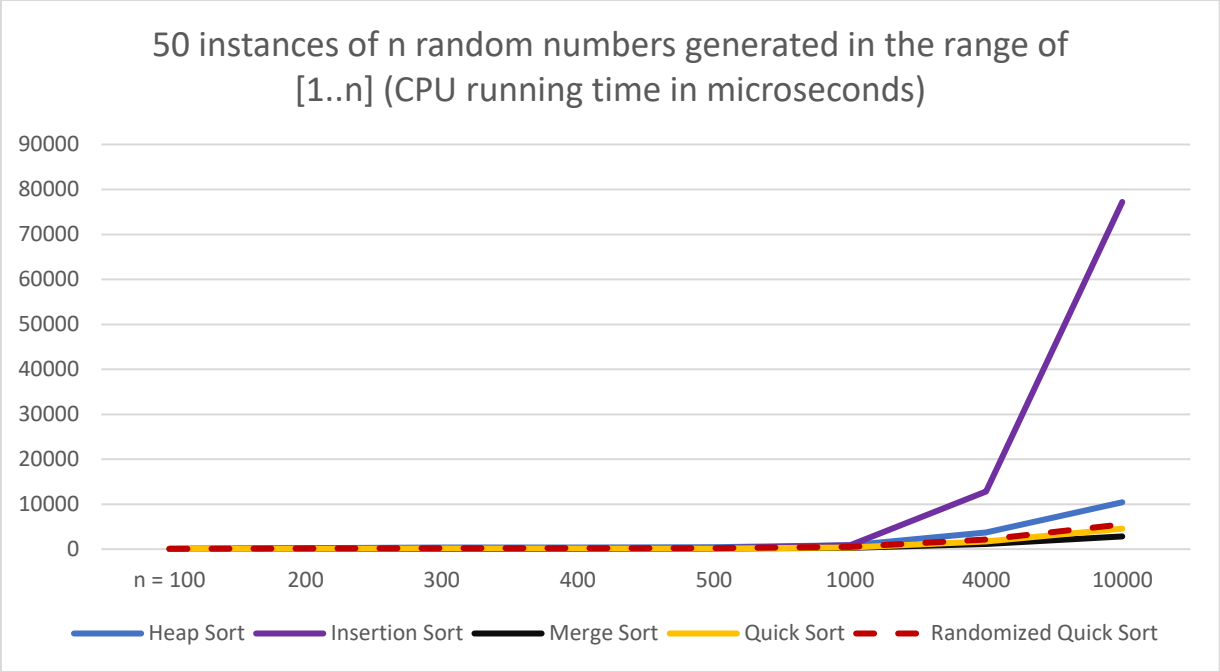


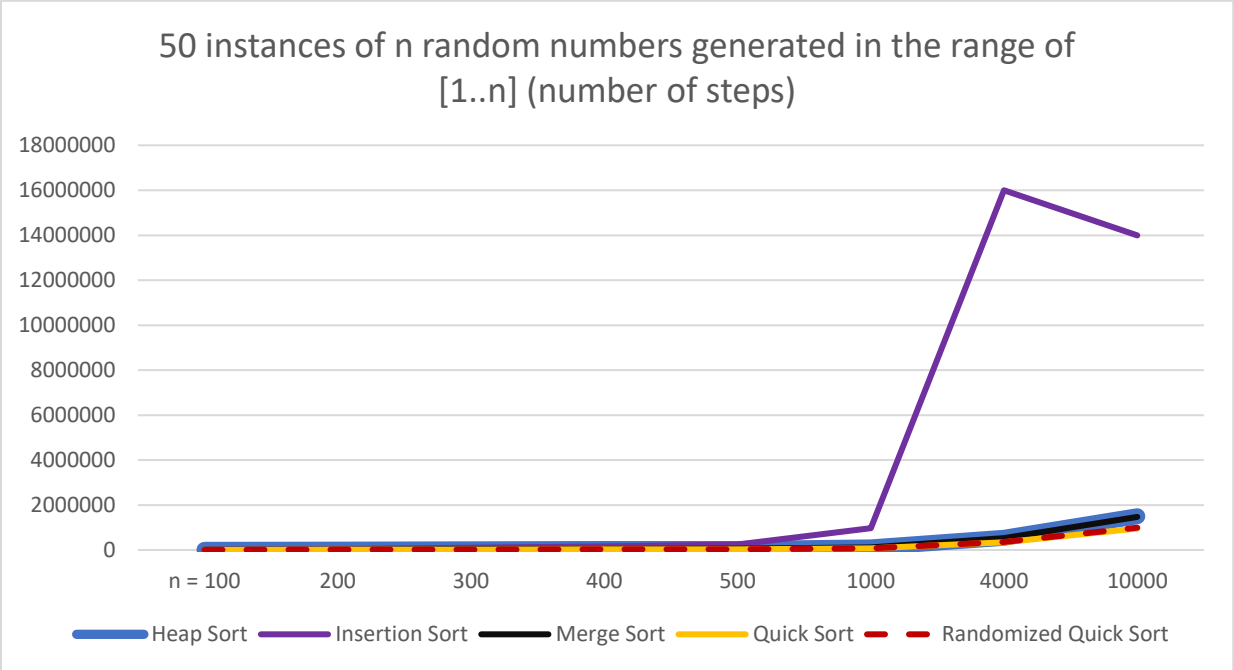


Ramiz Faragalla






50 instances of n random numbers generated in the range of [1..n] (CPU running time in microseconds)
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
n = 100
200
300
400
500
1000
4000
10000
Heap Sort
Insertion Sort
Merge Sort
Quick Sort
Randomized Quick Sort


50 instances of n random numbers generated in the range of [1..n] (number of steps)
18000000
16000000
14000000
12000000
10000000
8000000
6000000
4000000
2000000
0
n = 100
200
300
400
500
1000
4000
10000
Heap Sort
Insertion Sort
Merge Sort
Quick Sort
Randomized Quick Sort

Ramiz Faragalla

Approximation of constant C

Insertion sort: C = num steps / n^2

*Sorted input is excluded because it's O(n) in insertion sort

n	Reversely Sorted		Random Permutation		50 instances of n random numbers	
	Steps	C	Steps	C	Steps	C
100	20396	2.0396	12996	1.2996	10328	1.0328
200	80796	2.0199	51748	1.2937	37092	0.9273
300	181196	2.013288889	109304	1.214488889	94496	1.049955556
400	321596	2.009975	188952	1.18095	156344	0.97715
500	501996	2.007984	292044	1.168176	243448	0.973792
1000	2003996	2.003996	1118244	1.118244	968080	0.96808
4000	32015996	2.00099975	13013756	0.81335975	1.60323e+07	1.00201875
10000	200039996	2.00039996	84415316	0.84415316	1.40149e+07	0.38096448

Average C for insertion sort = (sum of c) / (8*3) = 32.341 / 24
Average C = 1.348

Merge sort: C = num steps / nlog(n)

n	Sorted		Reversely Sorted		Random Permutation		50 instances of n random numbers	
	Steps	C	Steps	C	Steps	C	Steps	C
100	7676	38.38	7596	37.98	8042	40.21	8050	40.25
200	17164	37.29634127	17004	36.94867088	18024	39.16506963	18078	39.28240839
300	27308	36.74695637	27052	36.40247047	28812	38.770811	28868	38.84616729
400	37940	36.45188824	37620	36.14443953	40182	38.6059508	40250	38.67128365
500	48436	35.89221067	48324	35.80921605	51592	38.23088061	51594	38.23236265
1000	105884	35.29466667	105660	35.22	112958	37.65266667	113252	37.75066667
4000	495572	34.3950407	494676	34.33285406	529700	36.76368531	532932	36.9880014
10000	1366932	34.1733	1358132	33.9533	1444832	36.1208	1.46991e+06	36.74775

Average C for merge sort = (sum of c) / (8*4) = 1187.70985898 / 32

Average C = 37.116

Ramiz Faragalla

Heap sort: C = num steps / nlog(n)

n	Sorted		Reversely Sorted		Random Permutation		50 instances of n random numbers	
	Steps	C	Steps	C	Steps	C	Steps	C
100	7650	38.25	6439	32.195	7143	35.715	7084.32	35.4216
200	17797	38.67181226	15049	32.70057328	16488	35.82743387	16542.8	35.94651098
300	28818	38.77888489	24723	33.26845621	26788	36.04721939	26874	36.16294512
400	40387	38.80291013	34949	33.57820354	37979	36.48935855	37853.9	36.36916532
500	52147	38.64214861	45527	33.73657353	49126	36.40351684	49145.5	36.4179668
1000	116446	38.81533333	102644	34.21466667	110744	36.91466667	110164	36.72133333
4000	564465	39.17654074	508453	35.28904302	536400	37.228697	536602	37.24271676
10000	1579911	39.497775	1428607	35.715175	1503151	37.578775	1.50271e+06	37.56775

Average C for heap sort = (sum of c) / (8*4) = 1165.38775184 / 32

Average C = 36.418

Randomized Quick sort: C = num steps / nlog(n)

n	Sorted		Reversely Sorted		Random Permutation		50 instances of n random numbers	
	Steps	C	Steps	C	Steps	C	Steps	C
100	4798	23.99	4678	23.39	5035	25.175	4906.26	24.5313
200	11008	23.91972295	10622	23.08096813	11833	25.71239841	11290.6	24.53379578
300	20118	27.07174704	18320	24.65227189	18662	25.11248351	18176.2	24.4587677
400	27006	25.94674997	24937	23.95890187	26537	25.49614545	25302.3	24.3098738
500	31180	23.10511043	34775	25.76908965	33082	24.51453699	32850.4	24.34291596
1000	71028	23.676	70381	23.46033333	76708	25.56933333	73882.4	24.62746667
4000	347666	24.12966475	353431	24.52978302	375210	26.04134862	355292	24.65894522
10000	995648	24.8912	967250	24.18125	926752	23.1688	991470	24.78675

Average C for randomized quick sort = (sum of c) / (8*4) = 786.79265447 / 32

Average C = 24.587