

Short Project 9: Divide and Conquer algorithms

Team No: 33

Vineet Vats: vxv180008

Yash Pradhan: ypp170130

Comparisons of the running times of the algorithms on randomly generated arrays:

- (a) Insertion sort
- (b) Merge sort (take 1)
- (c) Merge sort (take 2)
- (d) Merge sort (take 3)

Algorithm	Input Size(N)	Run Time (msec)	Memory (MB)
Insertion Sort	16	2	2
Insertion Sort	64	2	2
Insertion Sort	128	2	2
Insertion Sort	512	6	2
Insertion Sort	1000	7	2
Insertion Sort	10000	86	2
Insertion Sort	100000	4967	2
Insertion Sort	500000	96067	4
Insertion Sort	1000000	infinity	-
Merge Sort Take 1	1000	0	9
Merge Sort Take 1	10000	4	26
Merge Sort Take 1	100000	29	30
Merge Sort Take 1	500000	123	18
Merge Sort Take 1	1000000	258	23
Merge Sort Take 1	4000000	1176	31
Merge Sort Take 1	8000000	2476	62
Merge Sort Take 1	16000000	5114	203
Merge Sort Take 1	32000000	10712	374
Merge Sort Take 1	64000000	23642	629
Merge Sort Take 2	1000	0	2
Merge Sort Take 2	10000	2	6
Merge Sort Take 2	100000	20	11
Merge Sort Take 2	500000	81	14
Merge Sort Take 2	1000000	166	12
Merge Sort Take 2	4000000	824	672
Merge Sort Take 2	8000000	1774	519
Merge Sort Take 2	16000000	3850	672
Merge Sort Take 2	32000000	8177	738
Merge Sort Take 2	64000000	17968	738
Merge Sort Take 3	1000	0	2
Merge Sort Take 3	10000	1	6
Merge Sort Take 3	100000	18	11
Merge Sort Take 3	500000	83	14
Merge Sort Take 3	1000000	168	12
Merge Sort Take 3	4000000	850	672
Merge Sort Take 3	8000000	1862	519
Merge Sort Take 3	16000000	3692	672
Merge Sort Take 3	32000000	8169	738
Merge Sort Take 3	64000000	16247	738

Note:

1. Insertion Sort would take over 2 minutes for input size of about 1000000; hence marked as infinity
2. For the merge sort all versions; the time reported is the observed average of 100 trials.
3. When N was provided as over 64M; like 128M, We got out of memory exception, so we stopped there.