Oliver Fishstein

Project 3b

Overall, the Hash Table had the best performance because it allowed the shortest sort time and search time with the worst case (250) still being around 9 seconds faster than the next best, Quick Sort. The Quick Sort was the next fastest averaging around 0.1 seconds to sort with time taken up by the binary search. Heap Sort is the next fastest fall around 0.1 seconds behind Quick Sort. The next fastest was Merge Sort which was another 0.15 seconds behind Heap Sort. The slowest by far was Insertion Sort averaging around 11.25 seconds to sort.

Insertion Sort

Grid Size	Sort Time (sec)	Search Time (sec)	Total Time (sec)
10	186.2	0.017357	186.217
15	188.505	0.04175	188.547
30	187.932	0.169578	188.102
50	187.279	0.454475	187.734
250	183.984	11.5635	195.548

Quick Sort

Grid Size	Sort Time (sec)	Search Time (sec)	Total Time (sec)
10	0.10616	0.019872	0.125252
15	0.107029	0.045057	0.15211
30	0.109267	0.168475	0.277761
50	0.112763	0.455669	0.568451
250	0.106821	11.638	11.7448

Merge Sort

Grid Size	Sort Time (sec)	Search Time (sec)	Total Time (sec)
10	0.354989	0.017514	0.372526
15	0.368428	0.04869	0.401141
30	0.365095	0.172834	0.537952
50	0.359041	0.460104	0.819168
250	0.35644	11.5179	11.8743

Heap Sort

Grid Size	Sort Time (sec)	Search Time (sec)	Total Time (sec)
10	0.203985	0.017529	0.221537
15	0.200214	0.040419	0.240654
30	0.201983	0.170639	0.372644
50	0.205083	0.44767	0.652777
250	0.207742	11.4303	11.638

HashTable

Grid Size	Sort Time (sec)	Search Time (sec)	Total Time (sec)
10	0.044754	0.004531	0.049306
15	0.042136	0.009453	0.051613
30	0.042393	0.033241	0.075656
50	0.04768	0.098289	0.145992
250	0.044674	2.51517	2.55987