Sorting Techniques and Graph
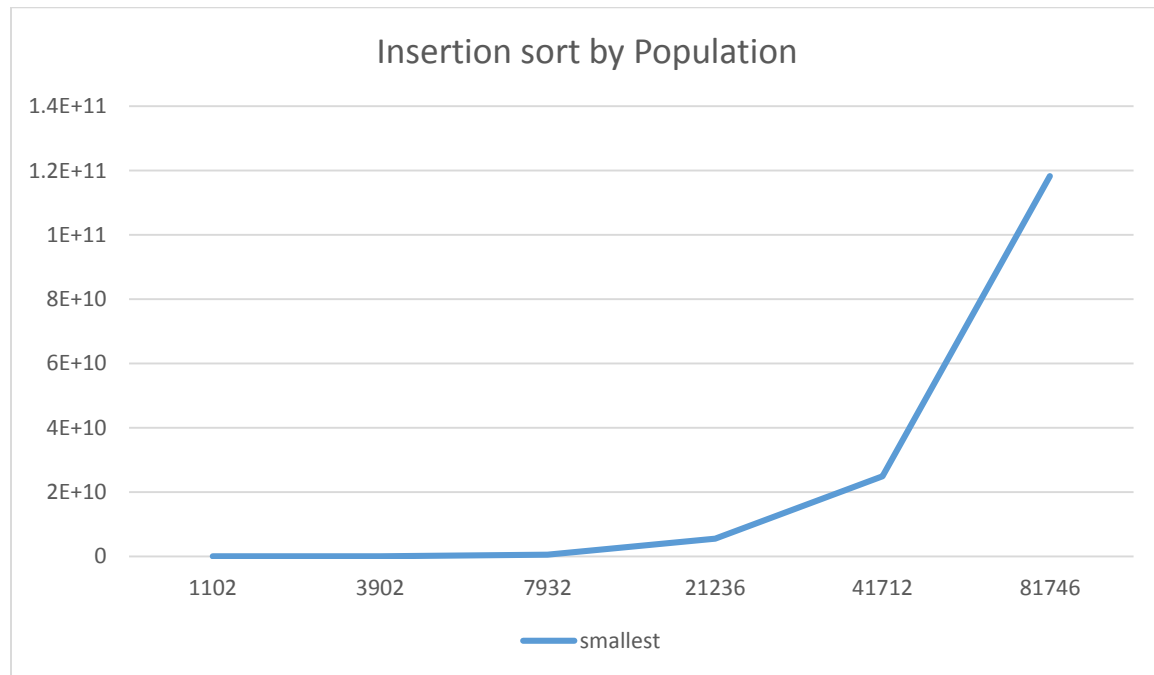
INSERTION SORT POPULATION

	CENSUS2010POP-Alabama-Alabama	CENSUS2010POP-Alabama-California	CENSUS2010POP-Alabama-Idaho	CENSUS2010POP-Alabama-Iowa	CENSUS2010POP-Alabama-Missouri	CENSUS2010POP
First	0:15545164	0:94330407	0:552254297	6:729250053	29:476733976	146:717996835
Second	0:10318074	0:117877723	0:553435688	5:986374156	40:714014576	158:794322239
Third	0:9241927	0:109825731	0:680566587	6:659836450	24:964683502	118:259392384
Fourth	0:17173122	0:112338263	0:603117117	5:517710317	34:402351470	134:916741956
Fifth	0:13567234	0:135571094	0:765284803	7:154675770	36:290028698	135:469602352
Smallest	0:10318074	0:94330407	0:552254297	5:517710317	24:964683502	118:259392384

Y-axis: CPU time in nanoseconds

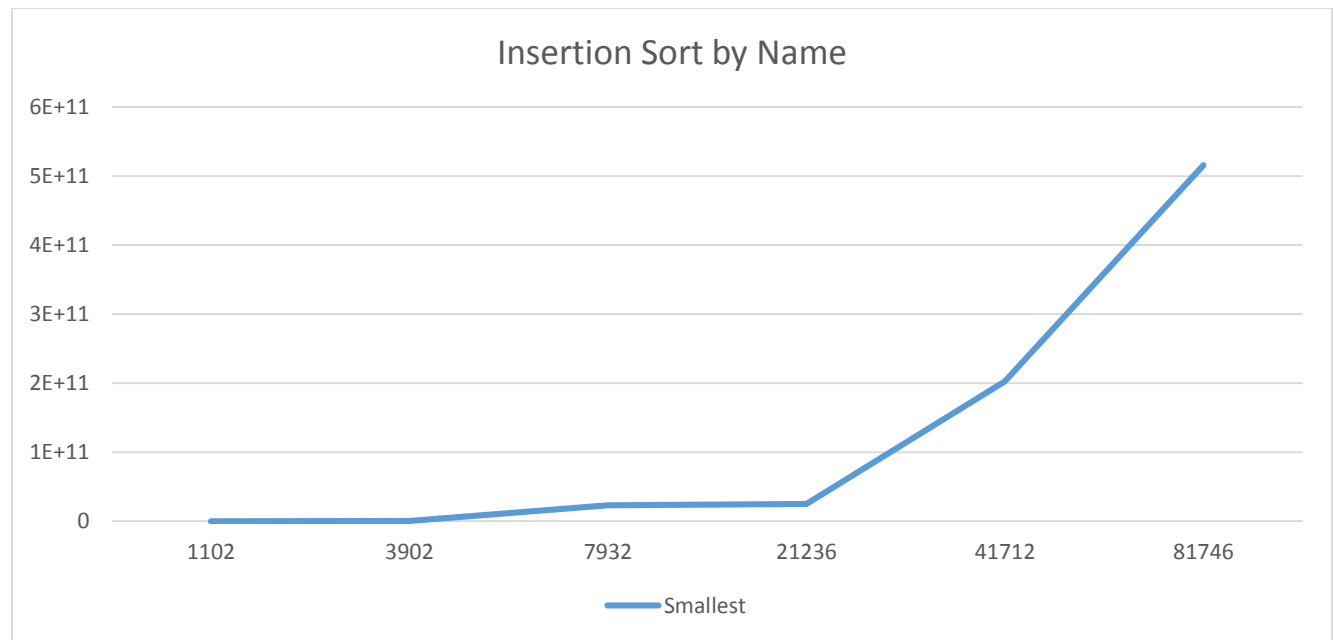X-axis: Number of records

-- smallest: smallest time taken to execute.

INSERTION SORT NAME

	CENSUS2010POP-Alabama-Alabama	CENSUS2010POP-Alabama-California	CENSUS2010POP-Alabama-Idaho	CENSUS2010POP-Alabama-Iowa	CENSUS2010POP-Alabama-Missouri	CENSUS2010POP
First	0:47268462	0:51427464 6	2:30768088 1	116:993827 439	20:2552630	551:249324170
Second	0:39105132	0:60876155 68	2:51442815 8	2:51442815 8	135:8106292 53	495:108456840
Third	0:40303084	0:51257188 8	2:89413642 0	22:1254694 59	98:78923432 9	424:545429176
Fourth	0:43064102	0:59404245 2	2:44067704 9	30:1526330 76	114:7126761 00	502:955463028
Fifth	0:48707102	0:66947037 8	2:64597519 8	21:1018916 24	124:5465010 52	515:675594700
Smallest	0:39105132	0:51257188 8	2:30768088 1	2:51442815 8	20:2552630	515:675594700

Y-axis: CPU time in nanoseconds

X-axis: Number of records

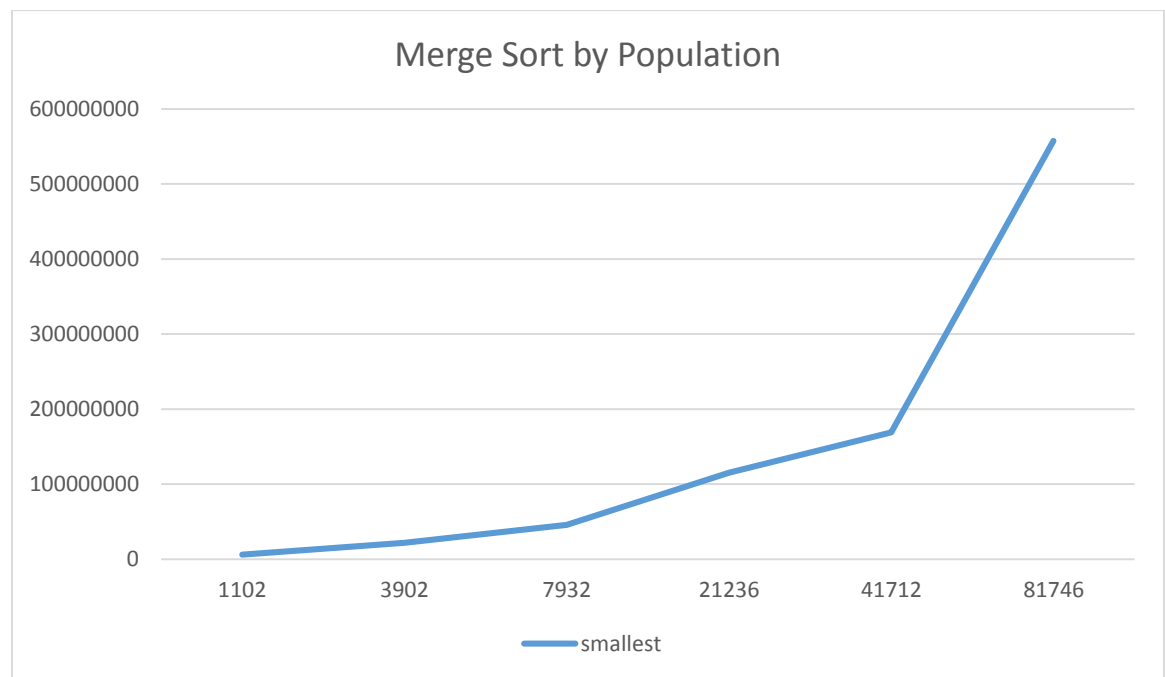--- smallest: smallest time taken to execute.

Insertion Sort by Name

MERGE SORT POPULATION

	CENSUS2010POP-Alabama-Alabama	CENSUS2010POP-Alabama-California	CENSUS2010POP-Alabama-Idaho	CENSUS2010POP-Alabama-Iowa	CENSUS2010POP-Alabama-Missouri	CENSUS2010POP
First	0:6332146	0:21749494	0:45816838	0:136511453	0:169144501	0:676924624
Second	0:6348666	0:26242828	0:45901569	0:115322413	0:435208096	0:597508101
Third	0:6004857	0:22145884	0:51224088	0:135860020	0:274899422	0:557605760
Fourth	0:6147377	0:22832730	0:45536530	0:229997165	0:315771014	0:664631929
Fifth	0:8704805	0:33318222	0:47794764	0:137271769	0:361787829	0:621182985
Smallest	0:6004857	0:21749494	0:45536530	0:115322413	0:169144501	0:557605760

Y-axis: CPU time in nanoseconds

X-axis: Number of records
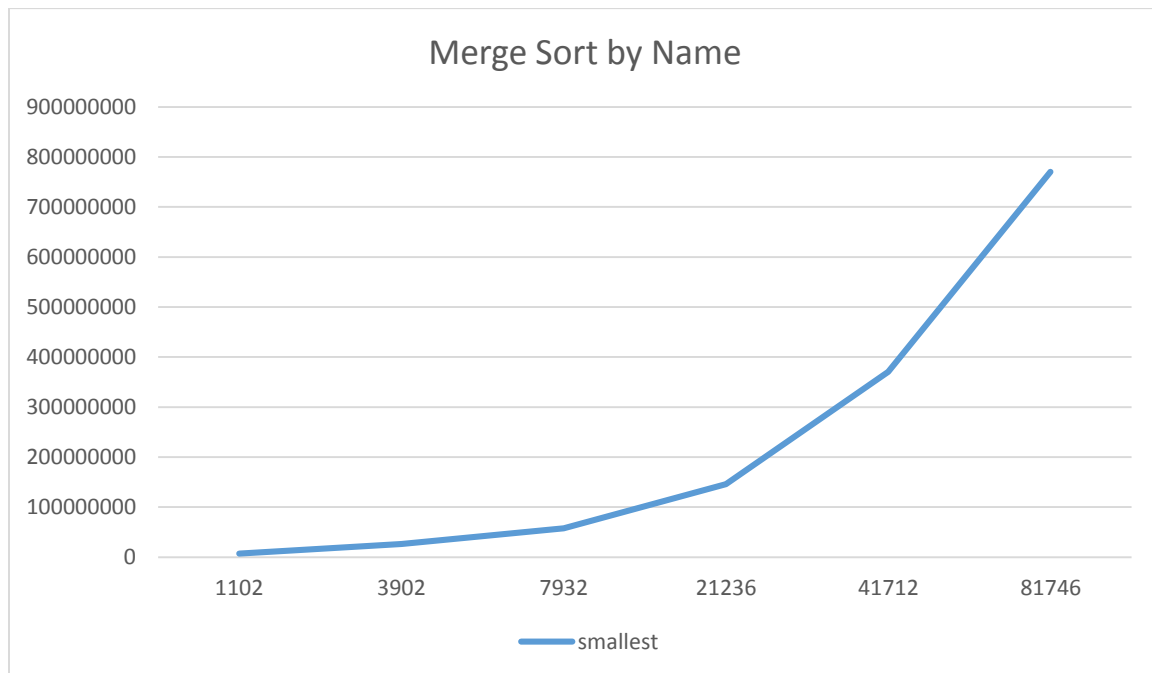
--- smallest: smallest time taken to execute.

Merge Sort by Population

MERGE SORT NAME

	CENSUS2010POP-Alabama-Alabama	CENSUS2010POP-Alabama-California	CENSUS2010POP-Alabama-Idaho	CENSUS2010POP-Alabama-Iowa	CENSUS2010POP-Alabama-Missouri	CENSUS2010POP
First	0:7132995	0:514274646	0:57535350	0:176790104	0:203729181	0:903801991
Second	0:8473572	0:32839530	0:62892000	0:146010544	0:487807738	0:865433155
Third	0:6973218	0:26617940	0:65938773	0:181831599	0:370810265	0:770230463
Fourth	0:7241590	0:28378465	0:60555795	0:248457997	0:404128644	1:151982088
Fifth	0:9527753	0:36987299	0:63840747	0:185512147	0:466878146	0:983132150
Smallest	0:6973218	0:26617940	0:57535350	0:146010544	0:370810265	0:770230463

Y-axis: CPU time in nanoseconds

X-axis: Number of records

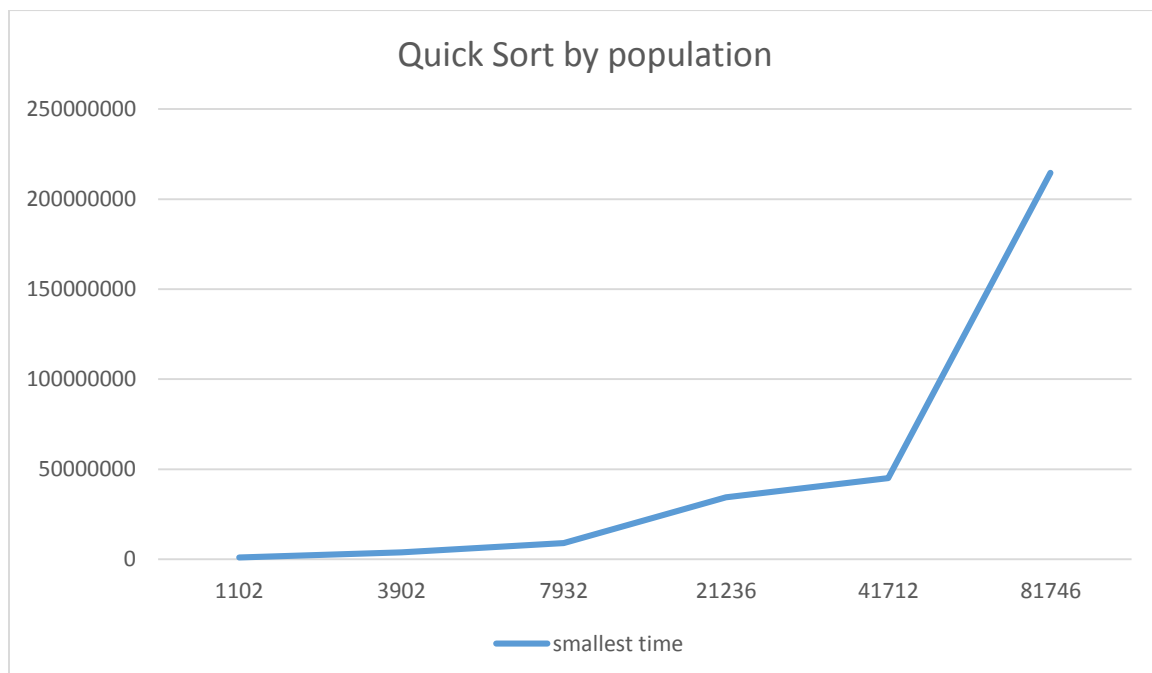--- smallest: smallest time taken to execute.

QUICK SORT POPULATION

	CENSUS2010POP-Alabama-Alabama	CENSUS2010POP-Alabama-California	CENSUS2010POP-Alabama-Idaho	CENSUS2010POP-Alabama-Iowa	CENSUS2010POP-Alabama-Missouri	CENSUS2010POP
First	0:1117043	0:3946976	0:9008335	0:34481086	0:45151577	0:422261906
Second	0:1082874	0:4341866	0:9742107	0:35058786	0:97382142	0:214604636
Third	0:6739358	0:25011185	0:81126472	0:169298246	0:377247259	0:790227468
Fourth	0:6672336	0:41665995	0:54406986	0:214729980	0:503674184	0:663511247
Fifth	0:8204105	0:33176673	0:57277859	0:186905798	0:481680685	1:113359589
Smallest	0:1082874	0:3946976	0:9008335	0:34481086	0:45151577	0:214604636

Y-axis: CPU time in nanoseconds

X-axis: Number of records

--- smallest: smallest time taken to execute.

QUICK SORT NAME

	CENSUS2010POP-Alabama-Alabama	CENSUS2010POP-Alabama-California	CENSUS2010POP-Alabama-Idaho	CENSUS2010POP-Alabama-Iowa	CENSUS2010POP-Alabama-Missouri	CENSUS2010POP
First	0:3421568	0:9245581	0:23860366	0:74278932	0:108141181	0:564948098
Second	0:2222712	0:16607143	0:22073159	0:126515496	0:276428106	0:444013870
Third	0:2074009	0:8854289	0:22493098	0:73605588	0:169937285	0:387148039
Fourth	0:2133025	0:19293398	0:21443490	0:104870538	0:237383326	0:297447507
Fifth	0:2613425	0:13565164	0:22038150	0:73296243	0:239046079	0:515482666
Smallest	0:2074009	0:13565164	0:21443490	0:73296243	0:108141181	0:297447507

Y-axis: CPU time in nanoseconds

X-axis: Number of records

--- smallest: smallest time taken to execute.
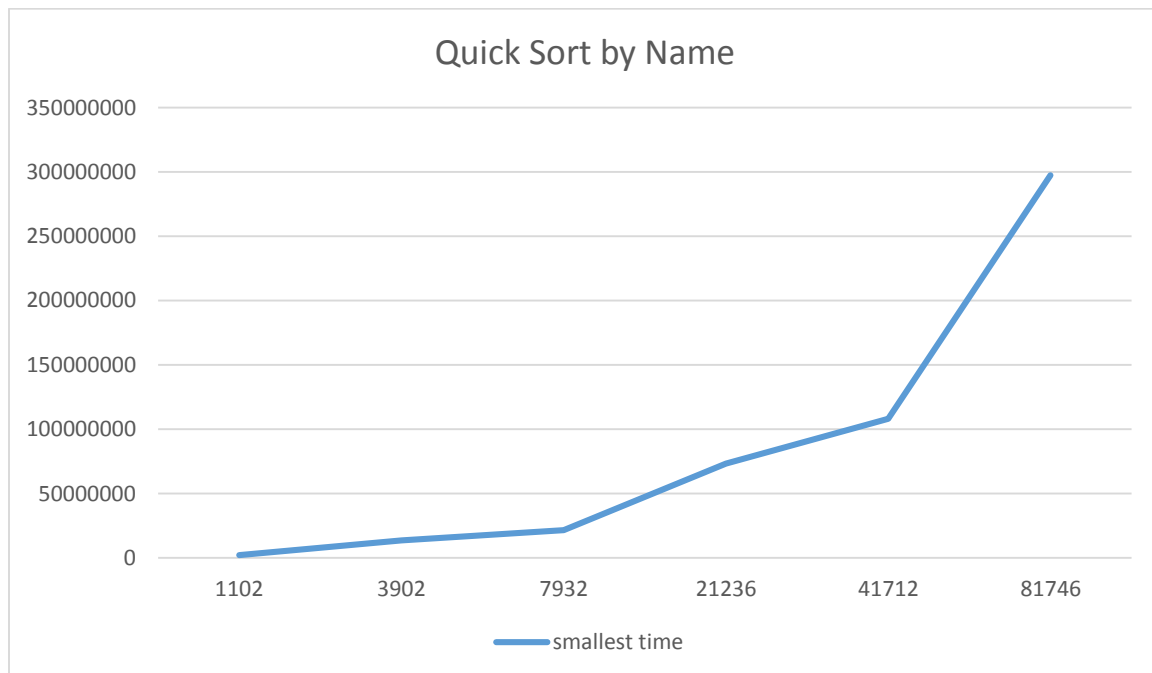
Conclusion: Smallest time required to sort 1102 records is taken by insertion sort (0:10318074 nanoseconds) among all the three sorts. Whereas smallest time to sort 81746 records is taken by Quicksort (0:214604636 nanoseconds). Maximum smallest time is taken by insertion sort to sort 81746 records that is (515:675594700 nanoseconds).Insertion sort takes linear time to sort small input size and quadratic time to sort large input size. Hence it is adaptive but not efficient when large data is to be considered. This states that for small records insertion sort can be used but should be avoided when large number of data is to be sorted. There is gradual increase in slope in case of merge and quick sort because it has same running time that is O (n logn) for average and best case whereas for insertion there is a steep slope as the running time becomes quadratic for 81746 records. Quicksort is an in-place sorting algorithm therefore comparison is done in the same vector. For merge sort we create two different vectors and sort accordingly. Thus quicksort takes less time than merge sort. Overall it takes less time to sort record by population than by name.