selection sort

#	Random	Ascending	Descending
50000	2928250	2530596	2731627
60000	4217830	3634821	3935534
70000	5743746	4956010	5359250
80000	7497989	6485603	6997772
90000	9484922	8199756	8835713
100000	11726152	10111550	10921941
110000	13829590	11956606	12883294

Bubble Sort

#	Random	Ascending	Descending
50000	4419667	191	3806293
60000	6379496	251	5486297
70000	8441816	294	7254168
80000	11367136	321	9761513
90000	14031026	383	11976432
100000	17812095	420	15177394
110000	20967007	485	17855939
500000		1927	
700000		2867	
1200000		4707	

Insertion

#	Random	Ascending	Descending
50000	1552793	268	3099826
60000	2202957	336	4470570
70000	2986246	360	4940399
80000	3870987	416	7936005
90000	4901478	463	9770110
100000	6073772	590	12499492
110000	7316057	2079	14568512
500000		3623	
700000		4460	
1200000		7054	

Merge

#	Random	Ascending	Descending
50000	7569	4252	4053
60000	9239	5076	4982
80000	13045	6819	6840
100000	16232	9055	8843
300000	52189	28484	28461
500000	89298	47968	49096
700000	127941	69842	69070
1200000	230116	122134	120290

Quick 1

#	Random	Ascending	Descending
50000	7579	3840	5603
60000	8955	4597	7204
80000	12112	6326	9443
100000	15469	8646	11984
300000	51471	25688	37675
500000	96842	43280	66812
700000	123619	64530	95930
1200000	218267	109615	171158

Quick 2

#	Random	Ascending	Descending
50000	7574	3513	4630
60000	9369	4212	5548
80000	11845	5671	7093
100000	15471	7401	9679
300000	51059	24393	30610
500000	85434	40517	52003
700000	123365	61175	76062
1200000	223048	107148	133532

Answer Question:

1. -**Random:** the selection sort, Bubble sort and insertion sort are quadratic growth rate, because I can observe that the curves of them are not linear from the graphs, and ttme increase whle the problem size increase s, their complexity is O(n^2) and the curves of them are above the merge, quick1 and quick2 sorts. Moreover, the merge sort, quick 1 and quick 2 sorts in the random condition are almost like N*logN, because the curves of them are below the selection sort, bubble and insertion sort, the time they (merge , quick1, quick2)used are smaller than others when the problem size is same. O(N*logN)<O(N^2)

Ascending: the selection will cost more time when the problem size is same, and its curve is quadratic growth rate, because the selection sort curve is above others a lot., and its curve is also quadratic growth rate, problem size increase s then time increases. On the other hand,the curve of bubble case in ascending is linear O(n). The insertion sort is llinear O(n) in ascending condition. because when the probelm size increase from N to 2N, the time also increase to 2T(previous time). Merge , quick 1 and quick2 sort are N*logN curve, because when the problem size is same, the time that these three sort used is larger than bubble and insertion , but lower than selection in ascending condition, then i can conclude that the its complexity must between O(N) and O(N^2), so it is O(N*logN).

Descending: the curves of selection, bubble, insertion sorts in descending are quadratic growth rate, O(n^2). because when problem size increase from N to 2N, the time increase to 4 times of previous time.
Merge , and quick 1 are n*logN curve, their curves are below the bubble and insertion when the problem size is same. And the quick 2 is also O(NlogN), because its curve is similar to quick 1.

2. N = 10,000, 000;

Selection sort: Random: n=50000, t = 2928250
N=10,000, 000, T = (10,000, 000/50000)^2 *t = 40000 *t ≈ 11.6*10^10
Ascending: n = 50000, t = 2530596
T = (10,000, 000/50000)^2 *t = 40000 *t ≈ 10.12*10^10
Descending: n = 50000, t = 2731627
T = (10,000, 000/50000)^2 *t = 40000 *t ≈ 10.92*10^10

Bubble sort: Random: n=50000, t = 4419667
T = (10,000, 000/50000)^2 *t = 40000 *t ≈ 17.6*10^10
Ascending: n=50000, t = 191
T = 10,000, 000/50000 *t = 200*t ≈ 38200
Descending: n=50000, t = 3806293
T = (10,000, 000/50000)^2 *t = 40000 *t ≈ 15.2*10^10

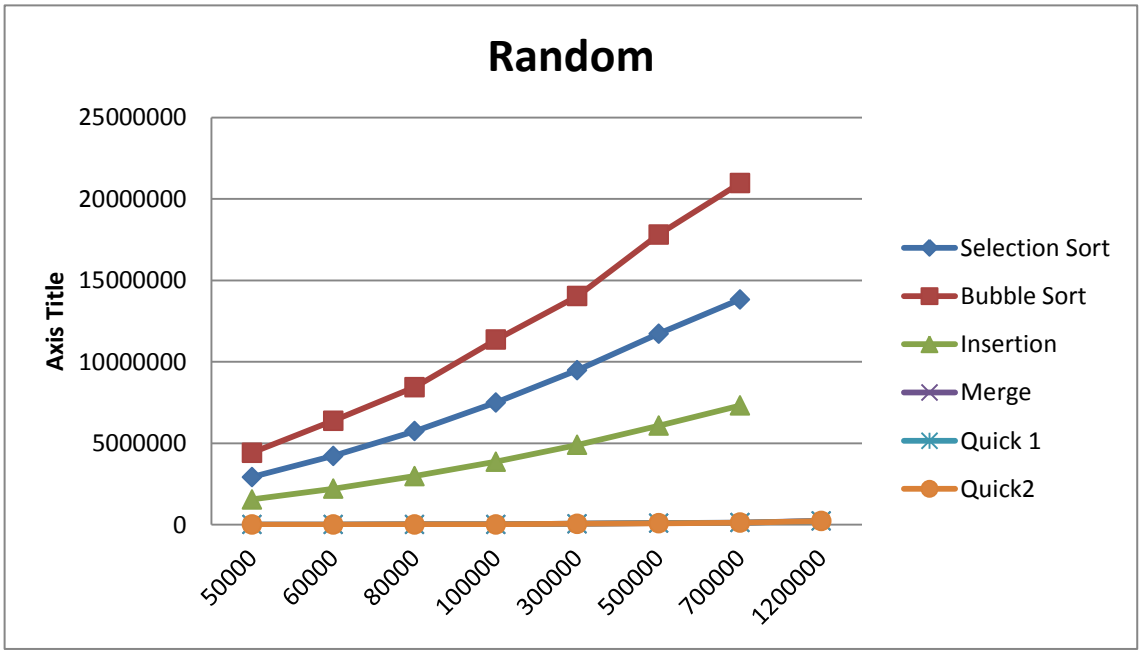
Insertion Sort: **:** Random: n=50000, t = 1552793
T = (10,000, 000/50000)^2 *t = 40000 *t ≈ 6.2*10^10
Ascending: n=50000, t = 268
T = 10,000, 000/50000 *t = 200*t ≈ 53600
Descending: n=50000, t = 3099826
T = (10,000, 000/50000)^2 *t = 40000 *t ≈ 12.36*10^10

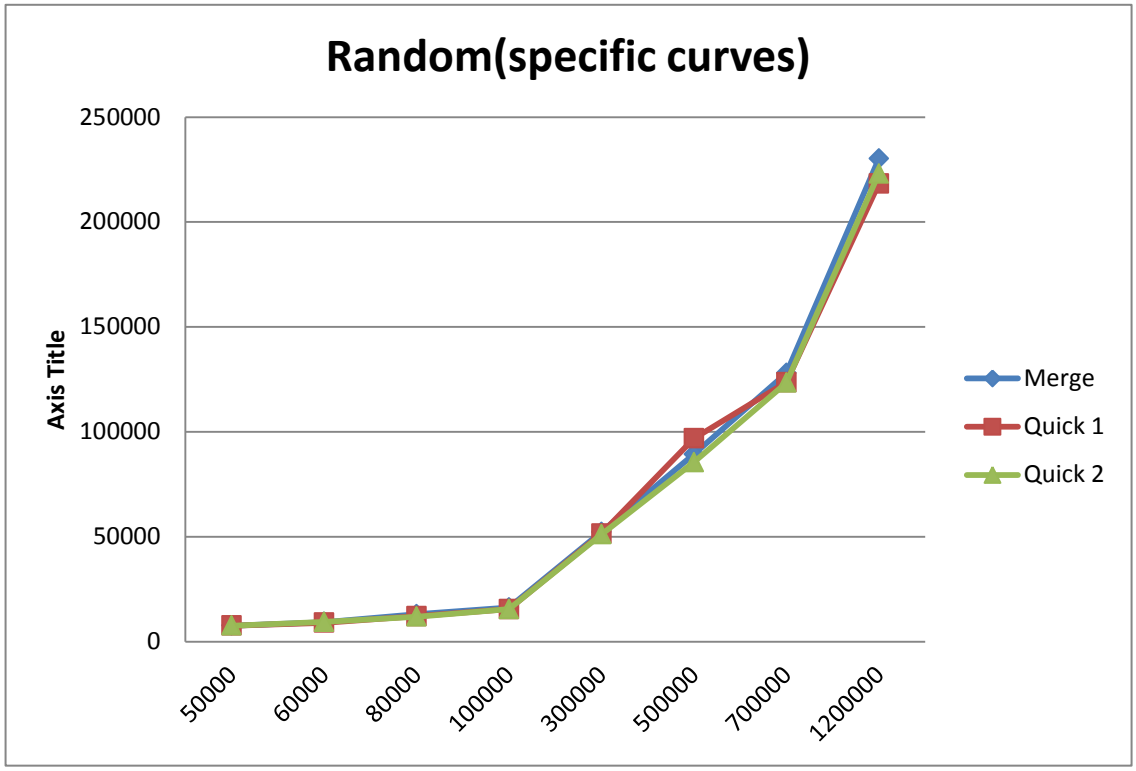
Merge Sort: **:** Random: n=50000, t =7569 , N = 10000000, T/t= NlogN/nlogn
O(NlogN) T ≈ 2107833
Ascending: n=50000, t =4252 , N = 10000000, T/t= NlogN/nlogn0
T ≈ 1131936
Descending: n=50000, t = 4053 ,T/t= NlogN/nlogn
T = 1123475

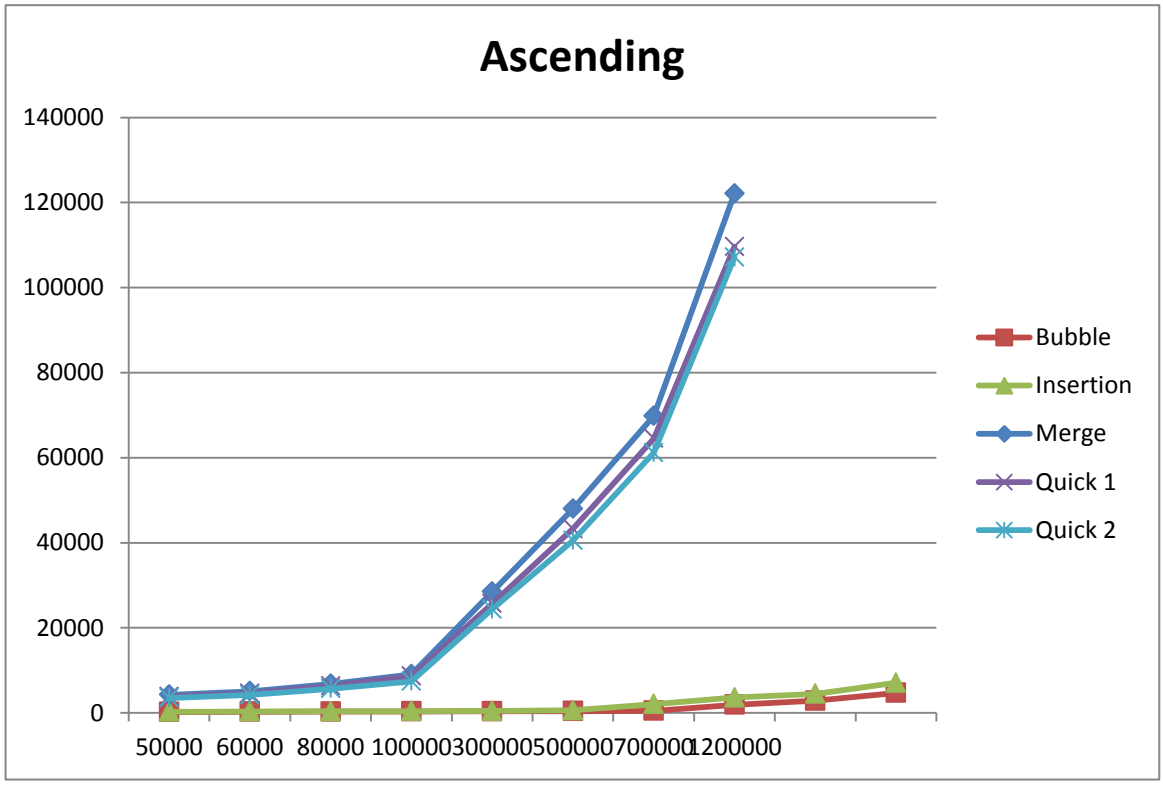
Quick1 Sort: **:** Random: n=50000, t = 7579,T/t= NlogN/nlogn
O(NlogN) T ≈ 1985671
Ascending: n=50000, t = 3840, T/t= NlogN/nlogn
T ≈ 1029164
Descending: n=50000, t = 5603 ,
T = 1578350

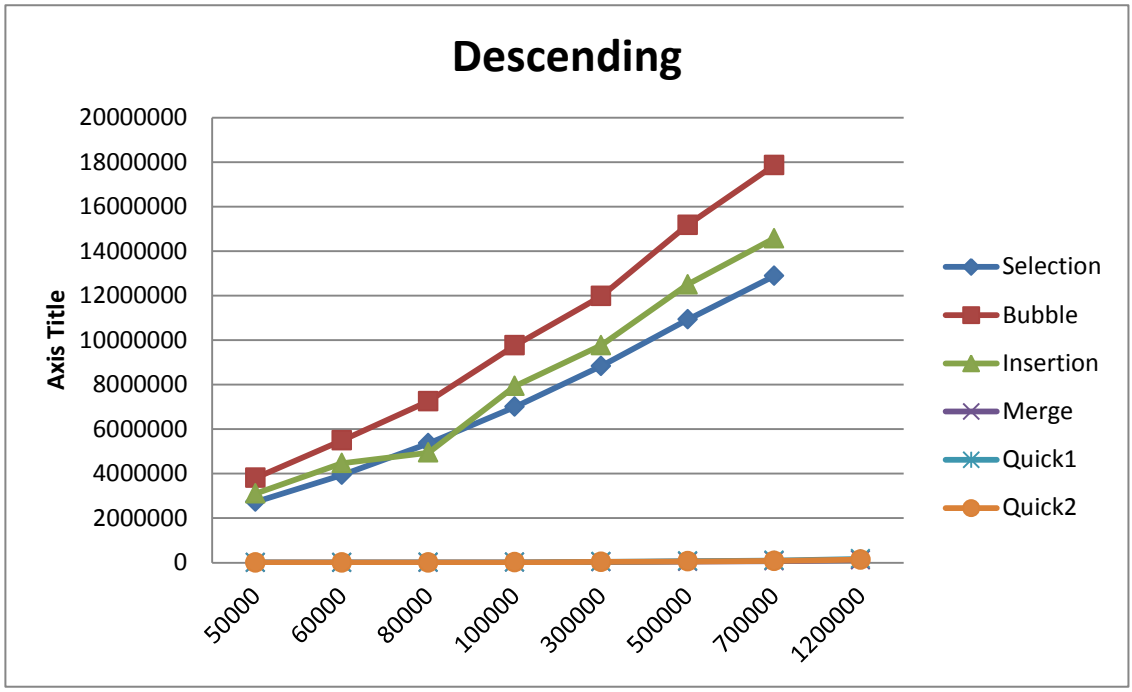
Quick2 Sort: **:** Random: n=50000, t = 7574
T ≈ 2018556
Ascending: n=50000, t = 3513
T ≈ 982299
Descending: n=50000, t =4630
T ≈ 1204447

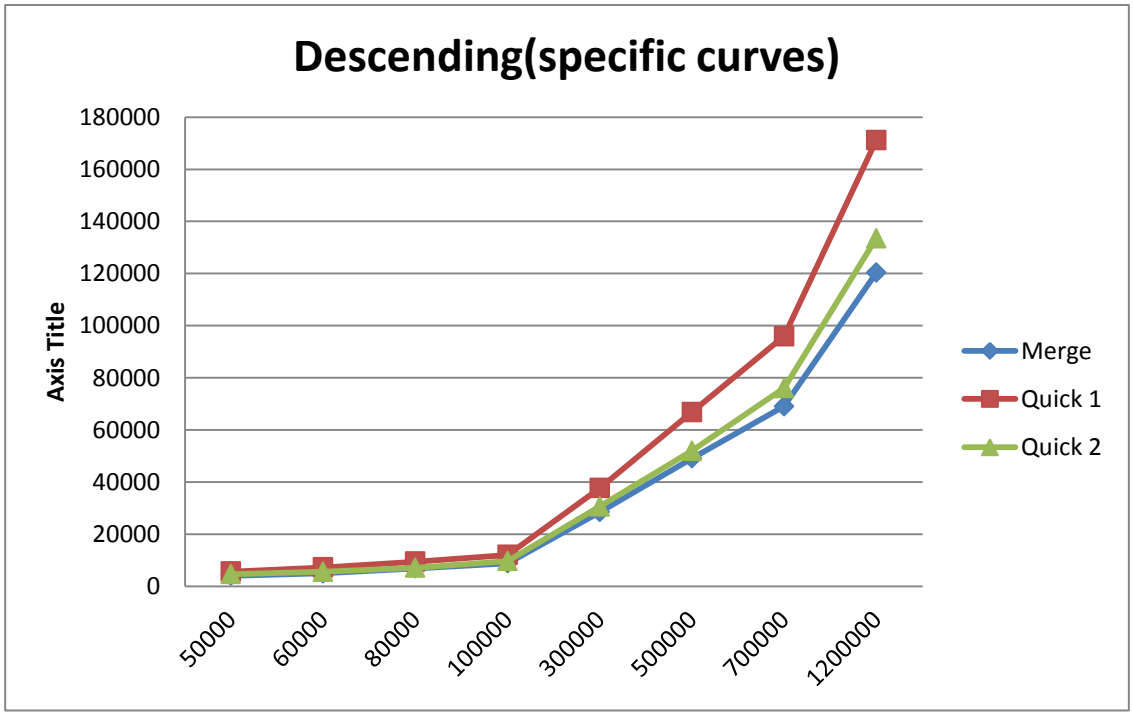

Selection Sort
16000000
14000000
12000000
10000000
8000000
6000000
4000000
2000000
0
Axis Title
50000
60000
70000
80000
90000
100000
110000
Random
Ascending
Descending


Random
25000000
20000000
15000000
10000000
5000000
0
Axis Title
50000
60000
80000
100000
300000
500000
700000
1200000
Selection Sort
Bubble Sort
Insertion
Merge
Quick 1
Quick2


Random(specific curves)
250000
200000
150000
100000
50000
0
Axis Title
50000
60000
80000
100000
300000
500000
700000
1200000
Merge
Quick 1
Quick 2


Ascending
140000
120000
100000
80000
60000
40000
20000
0
50000
60000
80000
100000
300000
500000
700000
1200000
Bubble
Insertion
Merge
Quick 1
Quick 2


Descending
20000000
18000000
16000000
14000000
12000000
10000000
8000000
6000000
4000000
2000000
0
Axis Title
50000
60000
80000
100000
300000
500000
700000
1200000
Selection
Bubble
Insertion
Merge
Quick1
Quick2


Descending(specific curves)
180000
160000
140000
120000
100000
80000
60000
40000
20000
0
Axis Title
50000
60000
80000
100000
300000
500000
700000
1200000
Merge
Quick 1
Quick 2