1. RESULTS

input_file	num_comp_qs	time_qs	num_comp_is	time_is
r20k	327078	0	100416650	0
r40k	706888	0	398333506	1
r60k	107139	0	901168796	2
r80k	147067	0	1603010026	3
r100k	1772227	0	2511601491	5
as20k	259009	0	258290	0
as40k	558072	0	334832	0
as60k	867305	0	556184	0
as80k	1196145	0	1086388	0
as100k	1534470	0	2094457	0

2. (1) For randomly generated data,
quick sort algorithm performs much faster!
random quick sort partition trees are quite good which in running time close to O(n log n). And it is an in-place sorting algorithm,which is to just generate the list of integers again, rather than comparing them.

(2) For almost sorted data,
I think the speed are quite close. and insertion sort algorithm uses more comparison. So Insertion sorting performs faster. But it is good only for sorting small arrays. The smaller the array, the faster insertion sort is compared to any other sorting algorithm

(3) For randomly generated data, quick sort with random pivot scheme is still the fastest one.
while for almost sorted data, it has to go through all the values in the array even its random pivot selected.

4When the elements are all the same, the running time of quick sort w value will be equivalent to the worst case running of quick sort. since no matter what pivot is picked,it will have to go through all the values in the array.