Complexity Analysis:

Insertion Sort:

In case of insertion sort, for every i from 1 to n-1, we need to check whether a[i] can be inserted in the already sorted subarray a[0..i-1]. On average, half of the elements of the subarray are smaller than a[i] and the other half is larger. So, we need to iterate i/2 times for every i. The running time will turn out to be a quadratic function of input size. So, the complexity is $O(n^2)$.

In best case, the array is already sorted. a[i] will always stay on its initial position since the condition key < a[j] will always be true in first iteration (where j is from 0 to i-1). So, it will take constant time for every iteration of i. So, the complexity is O(n).

In worst case, the array is inversely sorted. We need to loop through the subarray a[0..i-1] since a[i] will be smaller than every element of the subarray. So, the number of iterations is i. The running time will turn out to be a quadratic function of input size. So, the complexity is $O(n^2)$.

Selection Sort:

In case of selection sort, for every i from 0 to n-2, we need to check if a[i] is the smallest element of the subarray a[i..n-1]. How much the array is sorted does not matter in determining the complexity of the algorithm because we will always loop through all the remaining elements a[i+1..n-1] to find the smallest number each time. So, the total number of iterations of the inner loop is 1+2+…+n-1 = n(n-1)/2 which is a quadratic function. So, time complexity of selection sort is $O(n^2)$ for all cases – best, average and worst.

Machine Configuration:

OS: Windows 10

Processor: Intel® Core™ i7-8565U CPU @ 1.80GHz 1.99 GHz

Installed memory (RAM): 8.00 GB (7.85 GB usable)

System type: 64-bit Operating System, x64-based processor

Data:

input size	running time (millisecond)					
	best case		average case		worst cse	
	insertion	selection	insertion	selection	insertion	selection
10	0.0005	0.0003	0.001	0.0012	0.0003	0.0004
100	0.0005	0.0105	0.0088	0.0131	0.011	0.0105
200	0.0007	0.0402	0.0525	0.0561	0.0416	0.0389
500	0.0015	0.2458	0.1786	0.3602	0.254	0.2353
1000	0.0068	1.0349	1.1546	2.2934	0.9219	0.8639
2000	0.0053	3.5452	4.6322	7.1344	3.7155	3.8367
5000	0.0144	23.4199	14.2445	22.9006	23.288	23.7701
10000	0.0398	91.6491	49.2278	90.9047	92.3907	95.0593


average case time complexity
running time (millisecond)
100
80
60
40
20
0
10
100
200
500
1000
2000
5000
10000
input size
insertion
selection


worst case time complexity
running time (millisecond)
100
80
60
40
20
0
10
100
200
500
1000
2000
5000
10000
input size
insertion
selection