# COMP2521 Sort Detective Lab Report

by Yinla Zhu(z5510974)

In this experiment, my main purpose is to differentiate algorithms in sortA and sortB functions. In order to differentiate, due to the black box situation, a graph with time complicity will be the necessary data that shows the variation between 2 sort methods.

The independent variable in this experiment will be the number of data and the dependent variable will be the time taken to run a set of numbers of data.

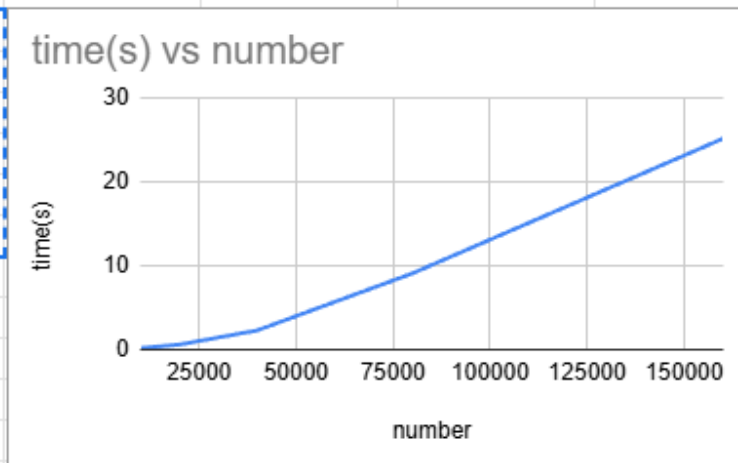| number	time(s)
10000	0.14
20000	0.56
40000	2.25
80000	9.06
160000	25.14
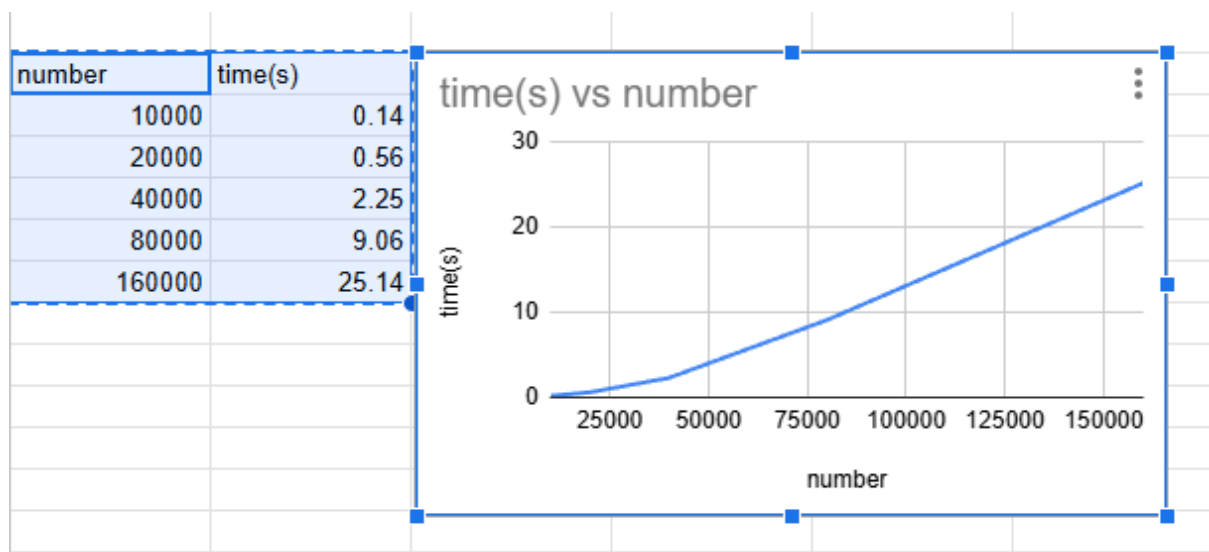
Due to the time issue, it takes forever time to run more than 160000 numbers of data, and according to these 5 time data, it shows a n^2 coefficient, for all 3 situation (random input, sorted input and reversed input) it will shorten the deduce range to 3 types of sorting algorithms below

1. bubble sort
2. selection sort
3. insertion sort

When we move on to sortB, the data I collected for the same independent variable were giving the same result for sorted input and reversed input

number	time(s)
10000	0.14
20000	0.56
40000	2.25
80000	9.06
160000	25.14

time(s) vs number

but the time taken to sort random inputs are shown as below

number	time(s)
1000000	0.45
2000000	0.94
3000000	1.38
4000000	1.88
5000000	2.66
6000000	2.91
7000000	3.45
8000000	3.94

time(s) vs number
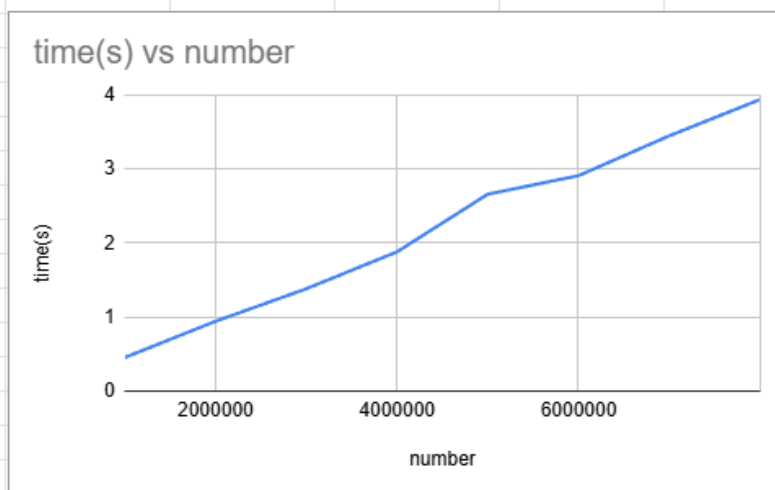
sortB algorithm with O(n) when random input and O(n^2)when sorted input will be either Insertion or Naive quicksort

To make sure the conclusion integrity, I will test stability for both sortA and sortB

result shows sortA is not stable

```
z5510974@vx13:~/2521/lab03$ rm asort_1
z5510974@vx13:~/2521/lab03$ rm asort_2
z5510974@vx13:~/2521/lab03$ sort -n < data > asort_1
z5510974@vx13:~/2521/lab03$ ./sortA < data > asort_2
z5510974@vx13:~/2521/lab03$ diff asort_1 asort_2
5d4
< 5 abc
6a6
> 5 abc
z5510974@vx13:~/2521/lab03$
```

in the range of sortA, the selection is the only algorithm that is not stable

result shows sortB is stable

```
z5510974@vx13:~/2521/lab03$ rm bsort_2
z5510974@vx13:~/2521/lab03$ sort -n < data > bsort_1
z5510974@vx13:~/2521/lab03$ ./sortB < data > bsort_2
z5510974@vx13:~/2521/lab03$ diff bsort_1
diff: missing operand after 'bsort_1'
diff: Try 'diff --help' for more information.
z5510974@vx13:~/2521/lab03$ diff bsort_1 bsort_2
z5510974@vx13:~/2521/lab03$
```

insertion is stable, so sortB will be insertion

Conclusion

On the basis of our experiments and our analysis above, we believe that

- sortA implements the *selection* sorting algorithm
- sortB implements the *insertion* sorting algorithm