On my honour, I pledge that I have neither received nor provided improper assistance in the completion of this assignment. 서명: __**강동인**___ 학번: __***21500002***____

Profiling Report

1. Profiling.exe captures

```
The maximum sample data size is 10000

    insertionsort(): sorted
         N   repetitions       sort(sec)
      1000        294845        0.000003
      2000        158604        0.000006
      3000        110864        0.000009
      4000         85150        0.000012
      5000         68943        0.000015
      6000         57331        0.000017
      7000         49669        0.000020
      8000         43648        0.000023
      9000         38871        0.000026
     10000         34748        0.000029

    insertionsort(): randomized
         N   repetitions       sort(sec)
      1000          1723        0.000580
      2000           440        0.002273
      3000           199        0.005045
      4000           111        0.009081
      5000            71        0.014141
      6000            50        0.020360
      7000            37        0.027703
      8000            28        0.036571
      9000            22        0.046182
     10000            18        0.056556

    insertionsort(): reversed
         N   repetitions       sort(sec)
      1000           877        0.001140
      2000           222        0.004505
      3000            99        0.010152
      4000            57        0.017789
      5000            36        0.028250
      6000            25        0.040400
      7000            19        0.054632
      8000            15        0.071267
      9000            11        0.091727
     10000             9        0.111667

    mergesort(): sorted
         N   repetitions       sort(sec)
```

```
  1000         19356        0.000052
  2000          9107        0.000110
  3000          5863        0.000171
  4000          4360        0.000229
  5000          3372        0.000297
  6000          2764        0.000362
  7000          2393        0.000418
  8000          2088        0.000479
  9000          1827        0.000547
 10000          1594        0.000627

mergesort(): randomized
     N   repetitions       sort(sec)
  1000         13900        0.000072
  2000          5224        0.000191
  3000          3248        0.000308
  4000          2357        0.000424
  5000          1831        0.000546
  6000          1493        0.000670
  7000          1247        0.000802
  8000          1084        0.000923
  9000           952        0.001051
 10000           841        0.001189

mergesort(): reversed
     N   repetitions       sort(sec)
  1000         19066        0.000052
  2000          9024        0.000111
  3000          5854        0.000171
  4000          4299        0.000233
  5000          3356        0.000298
  6000          2782        0.000359
  7000          2358        0.000424
  8000          2045        0.000489
  9000          1797        0.000556
 10000          1570        0.000637

quicksort(): sorted
     N   repetitions       sort(sec)
  1000           380        0.002637
  2000            95        0.010537
  3000            43        0.023651
  4000            24        0.042167
  5000            16        0.065937
  6000            11        0.095455
  7000             8        0.129500
  8000             6        0.169333
  9000             5        0.217600
 10000             4        0.266750

quicksort(): randomized
     N   repetitions       sort(sec)
  1000         23123        0.000043
  2000          8788        0.000114
  3000          4589        0.000218
  4000          2972        0.000336
  5000          2334        0.000428
  6000          1891        0.000529
```

```
      7000           1606         0.000623
      8000           1255         0.000797
      9000           1091         0.000917
     10000           1014         0.000986

quicksort(): reversed
         N   repetitions       sort(sec)
      1000           519         0.001931
      2000           119         0.008412
      3000            59         0.017017
      4000            35         0.028743
      5000            23         0.044261
      6000            16         0.063625
      7000            12         0.087333
      8000             9         0.111667
      9000             8         0.141250
     10000             6         0.174000
```

2. 성능 분석표

1. Insertion - Best

$$b = log\frac{T(2N)}{T(N)}$$

$$b = log\frac{t2(8000)}{t1(4000)} (t1(4000) \ = 0.000012,\ t2(8000) \ = 0.000023)$$

$$b \approx log_2 1.917$$

$$b \approx 0.939$$

$$T(4000) = a * 4000^{0.939}$$

$$0.000012 = a * 4000^{0.939}$$

$$a = 5.0 * 10^{-9}$$

a = 5.0 x 10^{-9}, b = 0.939

N	Time
10,000	0.000021 sec
20,000	0.000042 sec
1,000,000	Estimated: 0.002152 sec
	Measured: 0.002252 sec

$$Estimated:$$
$$T(1,000,000) \approx 5.0 * 10^{-9} * (1,000,000)^{0.939}$$
$$\approx 0.00215263305$$

2. Insertion - Average

$$b = log\frac{T(2N)}{T(N)}$$
$$b = log\frac{t2(8000)}{t1(4000)} (t1(4000) \ = 0.009081,\ t2(8000) \ = 0.036571)$$
$$b \approx log_2 4.027$$
$$b \approx 2.01$$

$$T(4000) = a * 4000^{2.01}$$
$$0.009081 = a * 4000^{2.01}$$
$$a = 5.2 * 10^{-10}$$

a = 5.2 x 10^{-10}, b = 2.01

N	Time
10,000	0.203800 sec
20,000	0.890500 sec
1,000,000	Estimated: 597.04 sec
	Measured: 538.127 sec

$$Estimated:$$
$$T(1,000,000) \approx 5.2 * 10^{-10} * (1,000,000)^{2.01}$$
$$\approx 597.039883178$$

3. Insertion - Worst

$$b = log\frac{T(2N)}{T(N)}$$
$$b = log\frac{t2(8000)}{t1(4000)} (t1(4000) \ = 0.017789,\ t2(8000) \ = 0.071267)$$
$$b \approx log_2 4.00624$$
$$b \approx 2.002$$

$$T(4000) = a * 4000^{2.002}$$
$$0.017789 = a * 4000^{2.002}$$
$$a = 1.1 * 10^{-9}$$

a = 1.1 x 10^{-9}, b = 2.002

N	Time
10,000	0.107400 sec
20,000	0.431000 sec
1,000,000	Estimated: 1124.15 sec
	Measured: 1077.63 sec

$$Estimated:$$
$$T(1,000,000) \approx 1.1 * 10^{-9} * (1,000,000)^{2.002}$$
$$\approx 1124.15819937$$

4. Quick - Average

$$b = log\frac{T(2N)}{T(N)}$$

$$b = log\frac{t2(8000)}{t1(4000)} (t1(4000) \ = 0.000336,\ t2(8000) \ = 0.0.000797)$$

$$b \approx log_2 2.372$$
$$b \approx 1.246$$

$$T(4000) = a * 4000^{1.246}$$
$$0.000336 = a * 4000^{1.246}$$
$$a = 1.1 * 10^{-8}$$

a = 1.1 x 10^{-8}, b = 1.246

N	Time
10,000	0.001171sec
20,000	0.002372sec
1,000,000	Estimated: 0.326 sec
	Measured: 0.1548sec

$$Estimated:$$
$$T(1,000,000) \approx 1.1 * 10^{-8} * (1,000,000)^{1.246}$$
$$\approx 0.326$$

5. Merge - Average

$$b = log\frac{T(2N)}{T(N)}$$

$$b = log\frac{t2(8000)}{t1(4000)} (t1(4000) \ = 0.000424, \ t2(8000) \ = 0.000923)$$

$$b \approx log_2 2.1769$$

$$b \approx 1.122$$

$$T(4000) = a * 4000^{1.122}$$

$$0.000424 = a * 4000^{1.122}$$

$$a = 3.8 * 10^{-8}$$

a = 3.8 x 10^{-8}, b = 1.122

N	Time
10,000	0.001337sec
20,000	0.002644sec
1,000,000	Estimated: 0.208 sec
	Measured: 0.165429sec

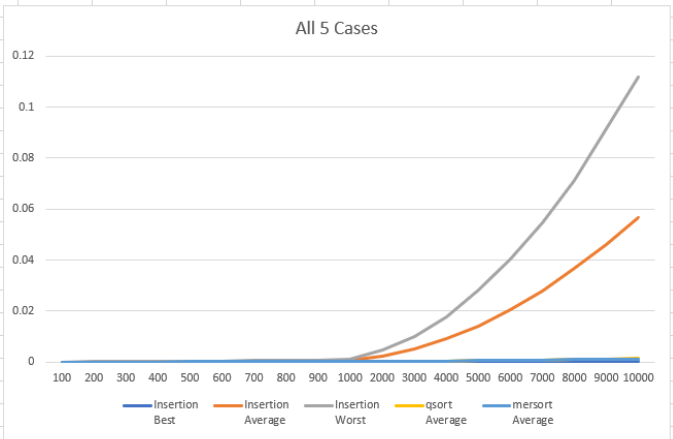
$$Estimated:$$

$$T(1,000,000) \approx 3.8 * 10^{-8} * (1,000,000)^{1.122}$$

$$\approx 0.20790002728$$

3. 5 cases graph

n	Insertion Best	Insertion Average	Insertion Worst	qsort Average	mersort Average
100	0	0.000006	0.000011	0.000003	0.000005
200	0.000001	0.000022	0.000042	0.000008	0.00001
300	0.000001	0.000048	0.000094	0.000011	0.000016
400	0.000001	0.000086	0.000166	0.000016	0.000022
500	0.000002	0.00013	0.000259	0.000021	0.000028
600	0.000002	0.000189	0.000373	0.000025	0.000035
700	0.000002	0.000256	0.000505	0.000034	0.000043
800	0.000002	0.000342	0.000658	0.000033	0.000051
900	0.000003	0.000433	0.000833	0.000038	0.000062
1000	0.000003	0.00058	0.00114	0.000057	0.000072
2000	0.000006	0.002273	0.004505	0.000114	0.000191
3000	0.000009	0.005045	0.010152	0.000238	0.000308
4000	0.000012	0.009081	0.017789	0.000337	0.000424
5000	0.000015	0.014141	0.02825	0.000457	0.000546
6000	0.000017	0.02036	0.0404	0.000543	0.00067
7000	0.00002	0.027703	0.054632	0.000679	0.000802
8000	0.000023	0.036571	0.071267	0.000927	0.000923
9000	0.000026	0.046182	0.091727	0.001036	0.001051
10000	0.000029	0.056556	0.111667	0.00125	0.001189



4. Time complexity & description

- selection sort

: bestcase 일 때는 매우 빠르지만 일반적인 경우와 worst case인 경우에 대해서는 asymptotic time complexity가 $O(N^2)$ 에 근접해서 매우 느려진다.

- merge sort

: N이 100만에 근접해도 best, worst, average 조건에 상관없이 빠르게 정렬되는 모습을 볼 수 있다.

- quick sort

: randomize 되어있을 때 좋은 효율을 보여줬고 worst cases일때는 $O(N^2)$ 에 근사한 시간복잡도를 보였다.