

PERFORMANCE EVALUATION OF SORTING

I added a global function for performance evaluation. It gets the size of the arrays as input and creates 5 different arrays with this size and performs tests on these arrays. I also added this function as one of the operations.

- 5 arrays of size 100

```
C:\Users\alper\Desktop\ODT\441\pa3\sorting\sort\bin\Debug\sort.exe
1. Initialize input array randomly
2. Load input array from a file
3. Perform Bubble Sort
4. Perform Quick Sort
5. Perform Selection Sort
6. Perform My Sort
7. Compare sorting algorithms
8. Delete and Finish
9. Performance Evaluation
9
What is the size of the arrays?
100
Algorithm      comparisons    moves          time(msec)
Bubble         10000          2528           0.04
Selection      10000          99             0.01
Quick          2696           96             0.02

Bubble         10000          2450           0.02
Selection      10000          99             0.02
Quick          2694           94             0.01

Bubble         10000          2472           0.04
Selection      10000          99             0.02
Quick          2694           94             0.02

Bubble         10000          2651           0.02
Selection      10000          99             0.02
Quick          2696           96             0.01

Bubble         10000          2445           0.04
Selection      10000          99             0.02
Quick          2694           94             0.01
```

- 5 arrays of size 1000

```
C:\Users\alper\Desktop\ODT\441\pa3\sorting\sort\bin\Debug\sort.exe
1. Initialize input array randomly
2. Load input array from a file
3. Perform Bubble Sort
4. Perform Quick Sort
5. Perform Selection Sort
6. Perform My Sort
7. Compare sorting algorithms
8. Delete and Finish
9. Performance Evaluation
9
What is the size of the arrays?
1000
Algorithm      comparisons    moves          time(msec)
Bubble         1000000       257362         1
Selection      1000000       999            2
Quick          251946        946            1

Bubble         1000000       256353         2
Selection      1000000       999            1
Quick          251962        962            1

Bubble         1000000       249287         2
Selection      1000000       999            2
Quick          251952        952            1

Bubble         1000000       248071         2
Selection      1000000       999            1
Quick          251942        942            1

Bubble         1000000       246972         2
Selection      1000000       999            1
Quick          251930        930            1
```

- 5 arrays of size 5000

```
C:\Users\alper\Desktop\ODT\441\pa3\sorting\sort\bin\Debug\sort.exe
1. Initialize input array randomly
2. Load input array from a file
3. Perform Bubble Sort
4. Perform Quick Sort
5. Perform Selection Sort
6. Perform My Sort
7. Compare sorting algorithms
8. Delete and Finish
9. Performance Evaluation
9

What is the size of the arrays?
5000
Algorithm      comparisons      moves      time(msec)
Bubble         25000000         6175385    110
Selection      25000000         4999       19
Quick          6259670          4670       7

Bubble         25000000         6256699    63
Selection      25000000         4999       19
Quick          6259662          4662       20

Bubble         25000000         6235976    120
Selection      25000000         4999       28
Quick          6259680          4680       5

Bubble         25000000         6298798    120
Selection      25000000         4999       23
Quick          6259668          4668       3

Bubble         25000000         6255521    69
Selection      25000000         4999       14
Quick          6259650          4650       17
```

- 5 arrays of size 10000

```
C:\Users\alper\Desktop\ODT\441\pa3\sorting\sort\bin\Debug\sort.exe
1. Initialize input array randomly
2. Load input array from a file
3. Perform Bubble Sort
4. Perform Quick Sort
5. Perform Selection Sort
6. Perform My Sort
7. Compare sorting algorithms
8. Delete and Finish
9. Performance Evaluation
9

What is the size of the arrays?
10000
Algorithm      comparisons      moves      time(msec)
Bubble         100000000        24952888   421
Selection      100000000        9999       117
Quick          25018800         8800       108

Bubble         100000000        24995736   366
Selection      100000000        9999       168
Quick          25018778         8778       117

Bubble         100000000        25077861   478
Selection      100000000        9999       131
Quick          25018838         8838       59

Bubble         100000000        25176826   349
Selection      100000000        9999       124
Quick          25018810         8810       60

Bubble         100000000        24682153   336
Selection      100000000        9999       126
Quick          25018784         8784       57
```

- 5 arrays of size 25000

```

C:\Users\alper\Desktop\ODT\441\pa3\sorting\sort\bin\Debug\sort.exe
1. Initialize input array randomly
2. Load input array from a file
3. Perform Bubble Sort
4. Perform Quick Sort
5. Perform Selection Sort
6. Perform My Sort
7. Compare sorting algorithms
8. Delete and Finish
9. Performance Evaluation
9

What is the size of the arrays?
25000
Algorithm      comparisons      moves      time(msec)
Bubble         625000000        156185403  2191
Selection      625000000        24999      700
Quick          156294048        19048      400

Bubble         625000000        156196208  2205
Selection      625000000        24999      691
Quick          156294098        19098      421

Bubble         625000000        156902045  2219
Selection      625000000        24999      668
Quick          156294012        19012      399

Bubble         625000000        155484944  2090
Selection      625000000        24999      663
Quick          156294098        19098      394

Bubble         625000000        156233149  2149
Selection      625000000        24999      663
Quick          156294046        19046      379

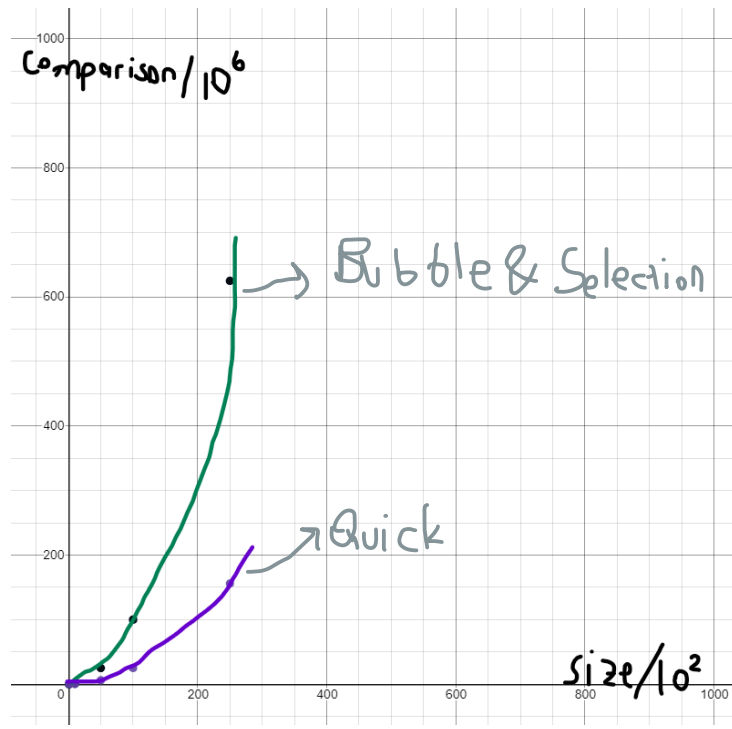
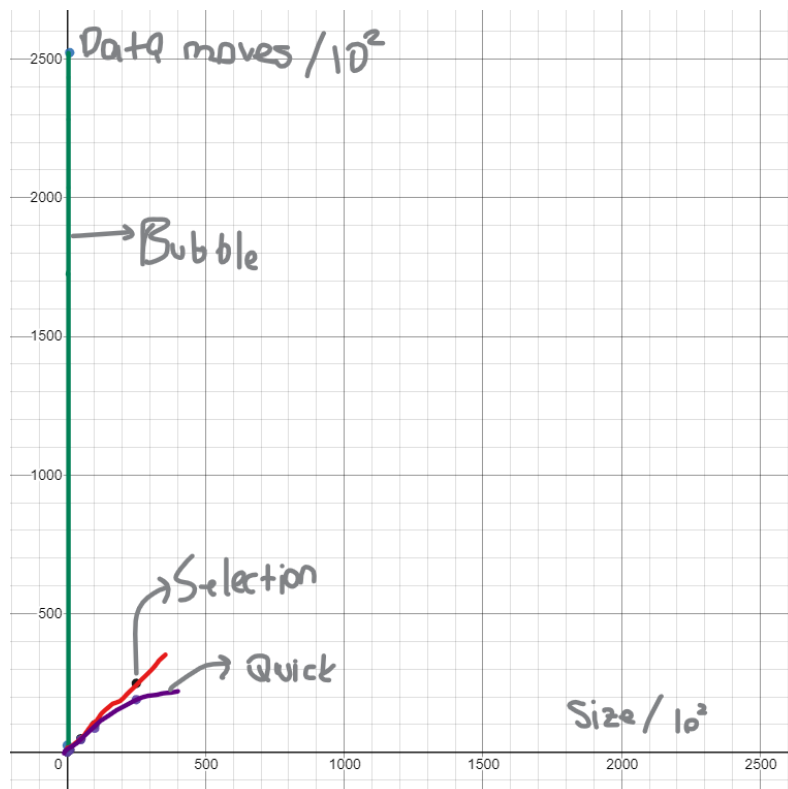
```

Tables:

Average # of comparisons	100	1000	5000	10000	25000
Bubble	10000	1000000	25000000	100000000	625000000
Selection	10000	1000000	25000000	100000000	625000000
Quick	2695	251948	6259678	25018802	156294060

Average # of data moves	100	1000	5000	10000	25000
Bubble	2509	252310	624012	24895673	156805642
Selection	99	999	4999	9999	24999
Quick	95	948	4670	8792	19050

Average elapsed time(msec)	100	1000	5000	10000	25000
Bubble	0.03	2	96	390	2170
Selection	0.02	1	20	133	677
Quick	0.01	1	10	80	399

Graphs:**Average # of comparisons****Average # of data moves**

Average elapsed time(msec)

