Key
1. Array size 100
2. Array size 1,000
3. Array size 10,000
4. Array size 100,000
5. Array size 250,000

3 Trials for Each Array Size

Bubblesort:
1. 0.0 s, 0.001 s, 0.001 s
2. 0.006 s, 0.004 s, 0.005 s
3. 0.156 s, 0.162 s, 0.161 s
4. 18.423 s, 18.582 s, 18.906 s
5. 141.316 s, 139.875 s, 138.239 s

Mergesort:
1. 0.002 s, 0.001 s, 0.0 s
2. 0.001 s, 0.001 s, 0.001 s
3. 0.002 s, 0.002 s, 0.002 s
4. 0.005 s, 0.004 s, 0.005 s
5. 0.007 s, 0.008 s, 0.009 s

Quicksort:
1. 0.001 s, 0.001 s, 0.001 s
2. 0.002 s, 0.002 s, 0.002 s
3. 0.005 s, 0.004 s, 0.004 s
4. 0.019 s, 0.018 s, 0.026 s
5. 0.032 s, 0.035 s, 0.046 s

Average Execution Time for Each Array Size

Bubblesort:
1. 0.000667 s
2. 0.005 s
3. 0.159667 s
4. 18.637 s
5. 139.81 s

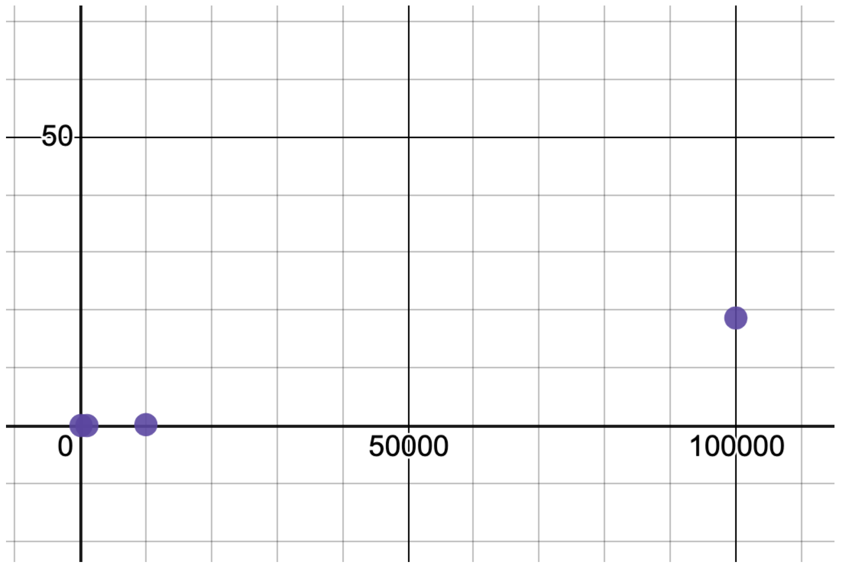
Mergesort:

1. 0.001 s
2. 0.001 s
3. 0.002 s
4. 0.004667 s
5. 0.008 s

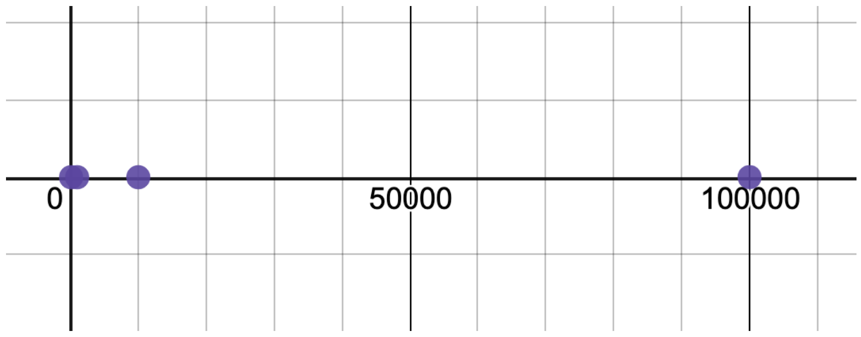
Quicksort:
1. 0.001 s
2. 0.002 s
3. 0.00433 s
4. 0.021 s
5. 0.037667 s

<u>Plots</u>

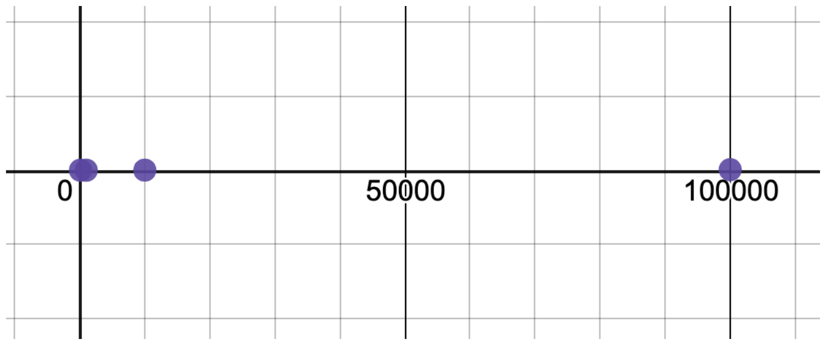
Bubblesort: Runtime is small for the smaller input sizes (100, 1000). However, as the input size increases, the runtime also increases. The runtime appears to increase exponentially.



Mergesort: Runtime remains relatively despite input size. The runtime does slightly increase as input size increases.



Quicksort: Very similar to Mergesort as runtime remains relatively small as input size increases. However, the runtime increases more than Mergesort when the input size reaches 100,000.



0
50000
100000