

Marc Ojalvo and Daniel Licht
CMPS 1500
Thursday 3:30-4:45
Lab 6 Part 3
03/21/2018

Running Times

- 10:
 - Merge sort:
 - It took 0.000132 seconds to input 10 values from file random10.txt
 - It took **5.7e-05 seconds** to sort 10 values using merge sort
 - It took 0.000531 seconds to output 10 sorted values to file orandom10.txt
 - Total time the program took is 0.00072 seconds
 - 92% of the running time is taken by input and output
 - Selection sort:
 - It took 0.000108 seconds to input 10 values from file random10.txt
 - It took **3.5e-05 seconds** to sort 10 values using selection sort
 - It took 0.000511 seconds to output 10 sorted values to file orandom10.txt
 - Total time the program took is 0.000654 seconds
 - 95% of the running time is taken by input and output
- 100:
 - Merge sort:
 - It took 0.000112 seconds to input 100 values from file random100.txt
 - It took **0.000518 seconds** to sort 100 values using merge sort
 - It took 0.000306 seconds to output 100 sorted values to file orandom100.txt
 - Total time the program took is 0.000936 seconds
 - 45% of the running time is taken by input and output
 - Selection sort:
 - It took 0.000108 seconds to input 100 values from file random100.txt
 - It took **0.000528 seconds** to sort 100 values using selection sort
 - It took 0.000402 seconds to output 100 sorted values to file orandom100.txt
 - Total time the program took is 0.001038 seconds
 - 49% of the running time is taken by input and output
- 1000:
 - Merge sort:
 - It took 0.000115 seconds to input 1000 values from file random1000.txt
 - It took **0.006542 seconds** to sort 1000 values using merge sort

- It took 0.000387 seconds to output 1000 sorted values to file random1000.txt
 - Total time the program took is 0.007044 seconds
 - 7% of the running time is taken by input and output
 - Selection sort:
 - It took 0.0001 seconds to input 1000 values from file random1000.txt
 - It took 0.04211 seconds to sort 1000 values using selection sort
 - It took 0.000563 seconds to output 1000 sorted values to file random1000.txt
 - Total time the program took is 0.042773 seconds
 - 2% of the running time is taken by input and output
- 10,000:
 - Merge sort:
 - It took 0.000137 seconds to input 10000 values from file random10000.txt
 - It took 0.067418 seconds to sort 10000 values using merge sort
 - It took 0.000336 seconds to output 10000 sorted values to file random10000.txt
 - Total time the program took is 0.067891 seconds
 - .70% of the running time is taken by input and output
 - Selection sort:
 - It took 0.000118 seconds to input 10000 values from file random10000.txt
 - It took 3.659305 seconds to sort 10000 values using selection sort
 - It took 0.000336 seconds to output 10000 sorted values to file random10000.txt
 - Total time the program took is 3.659759 seconds
 - .01% of the running time is taken by input and output
- 100,000:
 - Merge sort:
 - It took 0.000441 seconds to input 100000 values from file random100000.txt
 - It took 0.763385 seconds to sort 100000 values using merge sort
 - It took 0.000793 seconds to output 100000 sorted values to file orandom100000.txt
 - Total time the program took is 0.764619 seconds
 - .16% of the running time is taken by input and output
 - Selection sort:
 - It took 0.00035 seconds to input 100000 values from file random100000.txt
 - It took 362.39596 seconds to sort 100000 values using selection sort
 - It took 0.000813 seconds to output 100000 sorted values to file orandom100000.txt
 - Total time the program took is 362.397123 seconds

- .0003% of the running time is taken by input and output
- 1,000,000:
 - Merge sort:
 - It took 0.003143 seconds to input 1000000 values from file random1000000.txt
 - It took 8.900287 seconds to sort 1000000 values using merge sort
 - It took 0.002711 seconds to output 1000000 sorted values to file orandom1000000.txt
 - Total time the program took is 8.906141 seconds
 - .066% of the running time is taken by input and output
 - Selection sort:
 - Total time the program would take is less than 29 hours (104400 seconds). We stopped the program after about an hour and a half. We came up with this estimate because computers have gotten faster since 2014, when it took about 29 hours to sort 1,000,000 items with selection sort, and sort time for selection sort increases exponentially.

Short Answers:

1. The percentages are telling us that as the size of the list grows, the sorting process takes longer in proportion to the total run time.
2. The plots show that run time for merge sort is far shorter than run time for selection sort. Although with few elements, selection sort can be marginally faster than merge sort, as the size of the list grows, selection sort becomes extremely slow, due to the structure of the algorithm. Going through each element in order becomes much slower than dividing the list into lists of one element and then reordering the list from there into a sorted list.



