Week 8 CM1210
Exercises

19th March 2019

In the last lab we implemented selection sort in Java. We also counted the comparisons and swaps/iterations that would occur for different inputs to the algorithms.

In this lab we will start looking at how we can measure the execution time of these programs.

1 System.nanoTime()

The `System.nanoTime()` returns the current time in nanoseconds. Running this command directly before and after some code, and then subtracting the second time from the first, allows us to accurately work out how long that code took.

Time your selection sort code from the previous lab. You may want to create a copy of the code from the previous lab, and have your sorting methods free of any code for counting the number of swaps, etc. so that you're only timing the sorting code itself.

2 Multiple timings

To make the timings of your code more reliable, record the time taken to run the code multiple times (e.g. 10 times), and average it.

One thing that is important to note is that the ArrayList is changed when you pass it to the sorting method, so you may want to make a copy of the original ArrayList, and pass that copy to the sorting method, every time you perform the sort.

3 Varying the input

This section is more exploratory. To see what effect different inputs have on the sorting algorithm, try using larger/smaller ArrayLists, and vary the order of the elements in the ArrayList. Compare your results with what you were taught in lectures.