ICS4U Module 6: Note ↓ Exercise 3a

<u>Algorithm Analysis</u>

- *Algorithm Analysis* involves **measuring how efficiently an algorithm performs its task**
- A more efficient algorithm has a **shorter running time**
  - Running time is related to the number of statements executed to implement an algorithm
  - Can be estimated by calculating statement executions
  - Usually based on a worst-case set of data

Type of Sort/Search	Analysis for an array of n items
Selection Sort (uses nested for loops to sort items)	The outer for loop will be executed n times. The inner for loop will be executed n times. Total running time: $n * n = n^2$
Insertion Sort (while loop is used within a for loop)	Could allow for a faster sort in some cases, but in the worst case: The while loop will be executed $n - 1$ times. The for loop will be executed n times. Total running time: $n * (n - 1) \approx n^2$
Mergesort (more complicated divide and conquer algorithm)	Divide and conquer is more efficient than linear. Because this algorithm divides the array and each subarray in half until the base case of one element is reached, there are $log_2 n$ calls to mergesort() and then n calls to merge. Total running time: $nlog_2 n$
Binary Search (also divide and conquer)	Since elements are already ordered, only has to perform the search which involves dividing the array in half again and again. Total running time: $log_2 n$

<u>No Exercise for this Lesson!</u>