**10673261**

**Yeboah Perez Antwi**

***SUMMARY OF INSERTION SORT***

**Insertion sort** is an algorithm for sorting that builds the final array one item at a time. It is much less efficient on large lists than more advanced algorithms such as quicksort, merge sort, etc.

Insertion sort uses iteration, consuming one input element each repetition, and growing a sorted output list. At each iteration, insertion sort removes one element from the input data, finds the location it belongs within the sorted list, and inserts it there. It repeats until no input elements remain.

Sorting is typically done in-place, by iterating up the array, growing the sorted list behind it. At each array-position, it checks the value there against the largest value in the sorted list (which happens to be next to it, in the previous array-position checked). If larger, it leaves the element in place and moves to the next. If smaller, it finds the correct position within the sorted list, shifts all the larger values up to make a space, and inserts into that correct position.

The best case input is an array that is already sorted.  During each iteration, the first remaining element of the input is only compared with the right-most element of the sorted subsection of the array.

The simplest worst case input deals with an array sorted in reverse order. In these cases every iteration of the inner loop will scan and shift the entire sorted subsection of the array before inserting the next element.