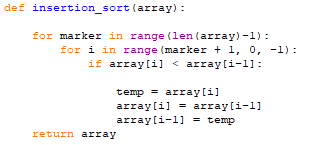
**Insertion Sort: (GCSE & A-Level)**

The Insertion sort inserts each item into its correct position in a data set one at a time,

*Simplified Explanation:*

1. Start at the second item in the list.
2. Compare current item with the first item in the sorted list
3. If the current item is greater than the item in the list, move to the next item in the sorted list.
4. Repeat from step 3 until the position of the current item is less than or equal to the item in the sorted list.
5. Move all the items in the list from the current position up one place to create a space for the current item.
6. Insert the current Item.
7. Repeat from step 2 with the next item in the list until all items have been inserted.

**Insertion Sort in Python:**



|  |  |
| --- | --- |
| **Advantages:** | **Disadvantages:** |
| * Can out perform a quicksort/bubblesort on small data sets | * It is less efficient on a large datasets as performance slows down |
| * It is a simple algorithm that is easy to make and implement |  |
| * Has the ability to sort a list as it is being received |  |

**Big O Notation: (A-Level)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Best Case:** | **Average Case:** | **Worst Case:** | **Space Complexity** |
| * Linear | * Polynomial | * Polynomial | * Constant |
| * O(n) | * O(n2) | * O(n2) | * O(1) |