**Code Design and Data Structures**

Assessable Exercise: Searching and Sorting

Nicole Dance (Brown)

Contents

[Task 1 – Sorting 3](#_Toc17119841)

[Bubble Sort Algorithm 3](#_Toc17119842)

[Why? 3](#_Toc17119843)

[Advantages 3](#_Toc17119844)

[Disadvantages 3](#_Toc17119845)

[Algorithm 3](#_Toc17119846)

[Why? 3](#_Toc17119847)

[Advantages 3](#_Toc17119848)

[Disadvantages 3](#_Toc17119849)

[Algorithm 3](#_Toc17119850)

[Why? 3](#_Toc17119851)

[Advantages 3](#_Toc17119852)

[Disadvantages 3](#_Toc17119853)

[Task 2 Searching 3](#_Toc17119854)

# Task 1 – Sorting

## Bubble Sort Algorithm

### Why?

Bubble sort was selected as one of my 3 sorting algorithms as it is one we have covered. Not only that but also due to the small amount of data in use for this assessment, Performance was not a major issue to consider. If I was to create an array of many more elements another algorithm would have bee preferred. One with increased performance.

### Advantages

* Simple to implement
* Works well with Arrays

### Disadvantages

* Slow due to comparing all elements against all other elements.
* Not very efficient

## Insertion Sort Algorithm

### Why?

Also, another algorithm covered. Again used as there isn’t a high number of elements in the array. Also whilst the same speed a bubble sort in its worst case. Otherwise, it is slightly faster. As it does not have to go through all of the previous sorted elements. But instead goes through them until it finds the correct spot to insert it.

### Advantages

* Quicker Then bubble sort in its best case due to swapping fewer elements

### Disadvantages

* Slow, And as the array grows it will take longer

## Algorithm

### Why?

### Advantages

### Disadvantages

# Task 2 Searching