**Searching and Sorting Algorithms**

**Question 1:**

An example of pseudocode for a Bubble Sort is:

**myList = list of items**

**#outerloop**

**FOR loop for length of myList -1:**

**#innerloop**

**FOR loop from 1 to length of myList:**

**if current item < item before it:**

**swap them**

1. Write pseudocode for a Selection/Simple Sort

**myList = list of items**

**FOR loop for length of myList:**

**find the next min value in list**

**if next min value < current value:**

**swap them**

1. Write pseudocode for an Insertion Sort

**myList = list of items**

**FOR loop from 2nd item to length of myList:**

**get current item**

**get position of current item**

**while sorted items > current item:**

**shift them to the right**

**insert current item in correct position**

**Question 2**

Use an insertion sort algorithm to put these European cities into alphabetical order

Note: An insertion sort is fairly straightforward but it is not sufficient to just take the list and rewrite it in order as that is not how a computer would do it. You should always show **every step** of the algorithm.

| **Riga** | **Paris** | **Oslo** | **Baku** | **Minsk** |
| --- | --- | --- | --- | --- |
| Paris | Riga | Oslo | Baku | Minsk |
| Oslo | Paris | Riga | Baku | Minsk |
| Baku | Oslo | Paris | Riga | Minsk |
| Baku | Minsk | Oslo | Paris | Riga |

**Question 3**

Six athletes compete in a long jump competition. Their best jumps are shown below

| 5.50 m | 5.32 m | 5.39 m | 6.50 m | 6.28 m | 6.14 m |
| --- | --- | --- | --- | --- | --- |

Show the stages of a bubble sort to list these jumps in order from shortest to longest. I have completed the first line for you.

Note: A common mistake is to forget the final pass because you realise that the list is already in order. You should always show a pass when **nothing changes** to complete the algorithm.

1st pass

5.50 5.32 5.39 6.50 6.28 6.14 Compare 5.32 and 5.50 and swap them

5.32 5.50 5.39 6.50 6.28 6.14 Compare 5.50 and 5.39 - swap

5.32 5.39 5.50 6.50 6.28 6.14 Compare 5.50 and 6.50 - no swap

5.32 5.39 5.50 6.50 6.28 6.14 Compare 6.50 and 6.28 - swap

5.32 5.39 5.50 6.28 6.50 6.14 Compare 6.50 and 6.14 - swap

5.32 5.39 5.50 6.28 6.14 6.50

2nd pass

5.32 5.39 5.50 6.28 6.14 6.50 Compare 5.32 and 5.39 - no swap

5.32 5.39 5.50 6.28 6.14 6.50 Compare 5.39 and 5.50 - no swap

5.32 5.39 5.50 6.28 6.14 6.50 Compare 6.28 and 6.14 - swap

5.32 5.39 5.50 6.14 6.28 6.50 Compare 6.28 and 6.50 - no swap

3rd pass

5.32 5.39 5.50 6.14 6.28 6.50 no swaps list sorted

**Question 4**

| 1992 | 2000 | 1989 | 1998 | 1990 |
| --- | --- | --- | --- | --- |

Show all the comparisons of a **bubble sort** algorithm to show that the list above is already in order.

1st pass

1992 2000 1989 1998 1990

1992 2000 1989 1998 1990

1992 1989 2000 1998 1990

1992 1989 1998 2000 1990

1992 1989 1998 1990 2000

2nd pass

1992 1989 1998 1990 2000

1989 1992 1998 1990 2000

1989 1992 1998 1990 2000

1989 1992 1990 1998 2000

3rd pass

1989 1992 1990 1998 2000

1989 1992 1990 1998 2000

1989 1990 1992 1998 2000

4th pass

1989 1990 1992 1998 2000

**Question 5**

Kim has a hand of playing cards, all of the same suit: 3,7,6,2,5

Arrange these cards in order from lowest to highest using an **insertion sort**

**3 7 6 2 5**

**3 7 6 2 5**

**3 6 7 2 5**

**2 3 6 7 5**

**2 3 5 6 7**