CCC2113 Data Structures and Algorithms Analysis **Name**:

Lab 7: Search Algorithms & List (30/5/2024) **Group**: A

1. Given the following unsorted array, show the subsequent steps for sorting using insertion sort algorithm.

Unsorted Array: Step 0

10	58	31	36	41	50	42	20

Step 1

10	58	31	36	41	50	42	20

Step 2

10	31	58	36	41	50	42	20

Step 3

10	31	36	58	41	50	42	20

Step 4

10	31	36	41	58	50	42	20

Step 5

10	31	36	41	50	58	42	20

Step 6

10	31	36	41	42	50	58	20

Step 7

10	20	31	36	41	42	50	58

2. What is a circular list? Give an example usage of a circular list?

A **circular list** is a type of linked list where the last node points back to the first node, creating a circular structure.

Ex:

Round-Robin Scheduling

In operating systems, circular lists are used to implement round-robin scheduling, where each process is assigned a fixed time slot in cyclic order.

Music Playlists

In music players, circular lists can be used to create playlists that repeat indefinitely, ensuring continuous playback without needing to restart the playlist manually.

3. With regard to algorithm complexity and stability, discuss why Quick sort is better than selection sort and shell sort?

Quick Sort generally outperforms Selection Sort and Shell Sort in terms of efficiency (average-case time complexity), making it more suitable for large datasets. Despite its worst-case scenario of O(n^2), which can be mitigated by good pivot selection strategies, its average-case performance and divide-and-conquer approach make it a better choice in practice. However, it is important to note that Quick Sort is not stable, so for applications requiring stability, other algorithms like Merge Sort might be preferable.