FAQ

Module-4	
Question 1.	**What distinguishes binary search from linear search in terms ofimportance?**
Answer	The linear search algorithm operates by iteratively examining each element ina sequential manner until either the desired target is found or the end of the data structure is reached. In contrast, the binary search algorithm partitions a sorted list into two equal halves iteratively, progressively removing half of theremaining possibilities until the desired target is located.
Question 2.	**Why is the semi-sorted or random beginning state of a list significantwhen selecting a sorting algorithm?**
Answer	The performance may be significantly influenced by the starting state. Certainalgorithms, such as bubble sort or insertion sort, exhibit adaptability by demonstrating improved performance when applied to partially sorted lists, but potentially experiencing longer execution times when confronted with entirely random collections.

Question 3.	How is the selection of a sorting algorithm impacted by the kind of storage (in-memory vs. sequential)?
Answer	The speed advantage of in-memory sorting may be attributed to the rapid access times offered by RAM. In the context of managing huge datasets stored sequentially, it is common practice to choose algorithms that minimise data transfer, such as merge sort or external sorting. This preference stems from the slower access times associated with secondary storage.
Question 4.	Are there circumstances in which linear search is preferable than binary search?
Answer	Indeed, in the case of tiny datasets or unsorted lists, the use of a linear search algorithm may prove to be more straightforward and comparably efficient. The use of binary search necessitates the prior sorting of the list, and thus may not exhibit superior performance in the case of very little datasets.

Question 5.	Why are sorting algorithms important in real-world settings like database administration or e-commerce?
Answer	The efficiency and user experience of a system are directly influenced by the use of sorting algorithms. In the context of electronic commerce, the implementation of a proficient search algorithm plays a crucial role in facilitating prompt product discovery for consumers, hence resulting in increased sales revenue. The efficiency of sorting operations in databases has a significant impact on the performance of query execution and data retrieval processes.