LAB 08 TASKS

Task # 01:

A courier dispatch company maintains a list of daily delivery records, where each record contains a Dispatch ID, Customer Name, and Delivery Time (in minutes). The manager wants to analyze the performance by sorting all dispatch records in ascending order of Delivery Time to identify the fastest and slowest deliveries. As a software developer, you are required to design a program that implements the Quick Sort algorithm to efficiently perform this task. Define a class DispatchRecord with data members int DispatchID, string CustomerName, and int DeliveryTime. Store multiple records in an array of DispatchRecord objects, then implement the Quick Sort algorithm with a partition function that selects the last element as the pivot and rearranges the records such that those with smaller DeliveryTime appear before the pivot and those with greater DeliveryTime appear after it. The program should display the list of records before sorting and again after sorting, clearly showing that the records are arranged in increasing order of delivery time.

Task # 02:

A regional bank stores customer account numbers as 7-digit integers in their daily transaction logs. Over time, these account numbers become unsorted due to the continuous addition of new transactions, which slows down searches during verification processes. The bank wants to improve efficiency by automatically organizing these account numbers in ascending order using Radix Sort. Write a program that inputs a list of 7-digit account numbers, applies Radix Sort by sorting digits from the least significant to the most significant place, and outputs the account numbers before and after sorting. The solution should clearly demonstrate how numbers are grouped and rearranged through each digit-processing stage.