



- **KEEPING MOBILE PHONE/ANY ELECTRONIC GADGETS, EVEN IN 'OFF' POSITION IS TREATED AS EXAM MALPRACTICE**
- **DON'T WRITE ANYTHING ON THE QUESTION PAPER**

Answer **ALL** Questions  
 (10 X 10 = 100 Marks)

- Write a program to record the temperatures of a city over 100 days. Count how many days fall within the range of 20°C to 30°C (mild days) and 35°C to 45°C (hot days) using a for loop and if-else conditions.
- Write a program to store n elements in an array and get a number to search. Your program should display whether the search number is present in the array or not.
- Identify the appropriate storage class by reading the given information and narrate the respective storage class with an example code
  - A storage class retains its value even after the function execution is complete.
  - The variables storage allocation is created only when the block is activated and perishes immediately after the execution.
- Is it possible for the Data structures to grow or shrink the size of the memory during runtime? Illustrate how the memory is managed with a suitable example.
- Write a program to create a structure **DEP\_STORE** and declare the variables as store\_name, store\_address, phone\_no. Create another structure **STORE\_ITEM** and declare the variables as item\_number, item\_name, available\_qty, item\_price. Generate a report to display the Store Name, Store Address, and Phone number with the item name and its price. Calculate the total item value present in the store. Refer to the example below :( STRUCTURE).

Item Number	ITEM NAME	Quantity	Price Per Item
101	LUNCH BOX	10	170
221	LAPTOP COVER	20	20
431	WALLET	20	305
<b>Total Item Value</b>			<b>6065</b>

6. Creating a library management system you need to design a class called **Book** with multiple constructors so that book objects can be initialized in different ways the various types of C++ constructor and how they are used. Provide suitable code to illustrate each type of constructor.
7. Design and implement a C++ program that fulfills the following requirements:
- Define a class that incorporates static data members and static member functions.
  - Illustrate how static data members are initialized and demonstrate the application of both static members and functions within the program with a suitable example.
8. We want to store the information of different vehicles. Create a class named **Vehicle** with two data members named mileage and price. Create its two subclasses **\*Car** with data members to store ownership cost, warranty (by years), seating capacity and fuel type (diesel or petrol). **\*Bike** with data members to store the number of cylinders, number of gears, cooling type(air, liquid or oil), wheel type(alloys or spokes) and fuel tank size(in inches) make another two subclasses **Audi** and **Ford** of **Car**, each having a data member to store the model type. Next, make two subclasses **Bajaj** and **TVS** of **car**, each having a data member to store the make-type. Now, store and print the information of an **Audi** and a **Ford** car (i.e. model type, ownership cost, warranty, seating capacity, fuel type, mileage and price.) Do the same for a Bajaj and a TVS bike.
- 9.a) Create a class **Time** to represent the departure time of a train, initialized with hours and minutes. Overload the pre-increment operator **++** to advance the departure time by one minute. Display the original and updated departure times.
- Note:** The time is represented in 24-hour format.

**OR**

- 9.b) Imagine you are designing a program for an e-commerce platform. Customers often search for items using different types of input parameters. For instance:
- i. A string input for searching by product name.
  - ii. An integer input for searching by product ID.
  - iii. A floating-point input for searching by price range.

To handle these diverse input types in an efficient way, you decide to use **function overloading** with **generic templates** to create a single search function.

10.a) Write a program to calculate the total cost of a meal for a group of people.

- The program has a base class, **MenuItem**, with an attribute price and a pure virtual function **calculatePrice()**.
- Two derived classes, **Starter** and **MainCourse**, inherit from **MenuItem**. They implement the pure virtual function to calculate the total cost based on the price of the item and the number of persons.

OR

10.b) A developer named Rohit Sharma is building a system that processes a list of integers provided by users. To ensure the data is clean, Rohit needs to filter out duplicate elements from the list and return a sorted list of unique integers. To accomplish this task, she uses the **Standard Template Library (STL)** list. The program should sort the list and remove duplicate elements, returning only unique numbers in sorted order.

↔↔↔ B/E/TY ↔↔↔