CHRIST (Deemed to be University), Bangalore – 560 029
Department of Computer Science
CIA - Component -II Practical Test – November 2024
SET -1
PG 2 Trimester

Programme Name: MCA   **Max. Marks: 30**

Course Name: Java Programming   **Time: 2 Hrs**

Course Code: MCA272

General Instructions

- All rough work should be done in the answer script. Do not write or scribble in the question paper except your register number.
- Verify the Course code / Course title & number of pages of questions in the question paper.
- Make sure your mobile phone is switched off and placed at the designated place in the hall.
- Malpractices will be viewed very seriously.
- Answers should be written on both sides of the paper in the answer booklet. No sheets should be detached from the answer booklet.
- Answers without the question numbers clearly indicated will not be valued. No page should be left blank in the middle of the answer booklet.
-

Course Outcomes (COs): The students will able to

CO1: Understanding and applying the principles of object-oriented programming in the construction of robust, maintainable programs.

CO2: Analyse the various societal and environmental problems critically to develop solutions using the features of programming language.

CO3: Develop sustainable and innovative solutions for real-time problems.

Answer all the questions: **Total Marks: 30**

Q. No	Questions	CO	RBT
1	**Marks 15** A government agriculture agency wants to simulate an application that tracks the functioning of agricultural tools and their interaction with various crops. Implement the following classes and interfaces: **Tool** *(Abstract Class)*: ○ Private attribute: toolName (String) ○ Protected method: setToolName(String name) ○ Public method: getToolName() **Interface Maintenance**: ○ Method: void performMaintenance() **Interface CropManagement**: ○ Method: void manageCrop(String cropType) **Class Machine extends Tool implements Maintenance**:	CO2 & CO3	L3, L4 & L5

	○ Attribute: machineType (String) ○ Constructor to initialize toolName and machineType ○ Override performMaintenance() method to print a message about maintaining the machine **Class Tractor extends Machine implements CropManagement**: ○ Attribute: fuelType (String) ○ Constructor to initialize toolName, machineType, and fuelType ○ Override manageCrop(String cropType) method to print a message about managing the given crop type using the tractor **Task:** Write the full implementation for the above classes and interfaces in Java, ensuring proper utilization of **private**, **protected**, and **public** access specifiers. Test the program with the following scenario: ● Create an instance of the Tractor class. ● Set the tool name using the setToolName method. ● Print the tool name and manage a crop ("Wheat") using the tractor. ● Call the performMaintenance method for the tractor.		
2	**SHARED DIGIT (7.5 Marks)** Write a method named **hasSharedDigit** with two parameters of type **int**. Each number should be within the range of **10 (inclusive) - 99 (inclusive)**. If one of the numbers is **not within the range**, the method should return false. The method should return true if there is a digit that appears in both numbers, such as **2** in 12 and 23; otherwise, the method should **return false.** **EXAMPLE INPUT/OUTPUT:** **hasSharedDigit(12, 23);** → should **return true** since the digit **2** appears in both numbers **hasSharedDigit(9, 99);** → should **return false** since **9** is **not within the range of 10-99** **hasSharedDigit(15, 55);** → should **return true** since the digit **5** appears in both numbers **NOTE:** The method **hasSharedDigit** should be defined as **public static**	CO1, CO2 & CO3	L3, L4 L5, L6
3	**Playing Cat (7.5 Marks)** The cats spend most of the day playing. In particular, they play if the temperature is between 25 and 35 (inclusive). Unless it is summer, then the upper limit is 45 (inclusive) instead of 35.		

Write a method **isCatPlaying** that has **2 parameters.** Method needs to return true if the cat is playing, otherwise return false

1st parameter should be of type **boolean** and be named **summer** it represents if it is summer.

2nd parameter represents the **temperature** and is of type int with the **name temperature**.

EXAMPLES OF INPUT/OUTPUT:

- **isCatPlaying(true, 10);** should return **false** since temperature is not in range 25 - 45
- **isCatPlaying(false, 36);** should return **false** since temperature is not in range 25 - 35 (summer parameter is false)
- **isCatPlaying(false, 35);** should return **true** since temperature is **in range 25 - 35**

Revised Bloom's Taxonomy (RBT) Levels :		
L1 – Remembering	L2 – Understanding	L3 – Applying
L4 – Analyzing	L5 – Evaluating	L6 - Creating