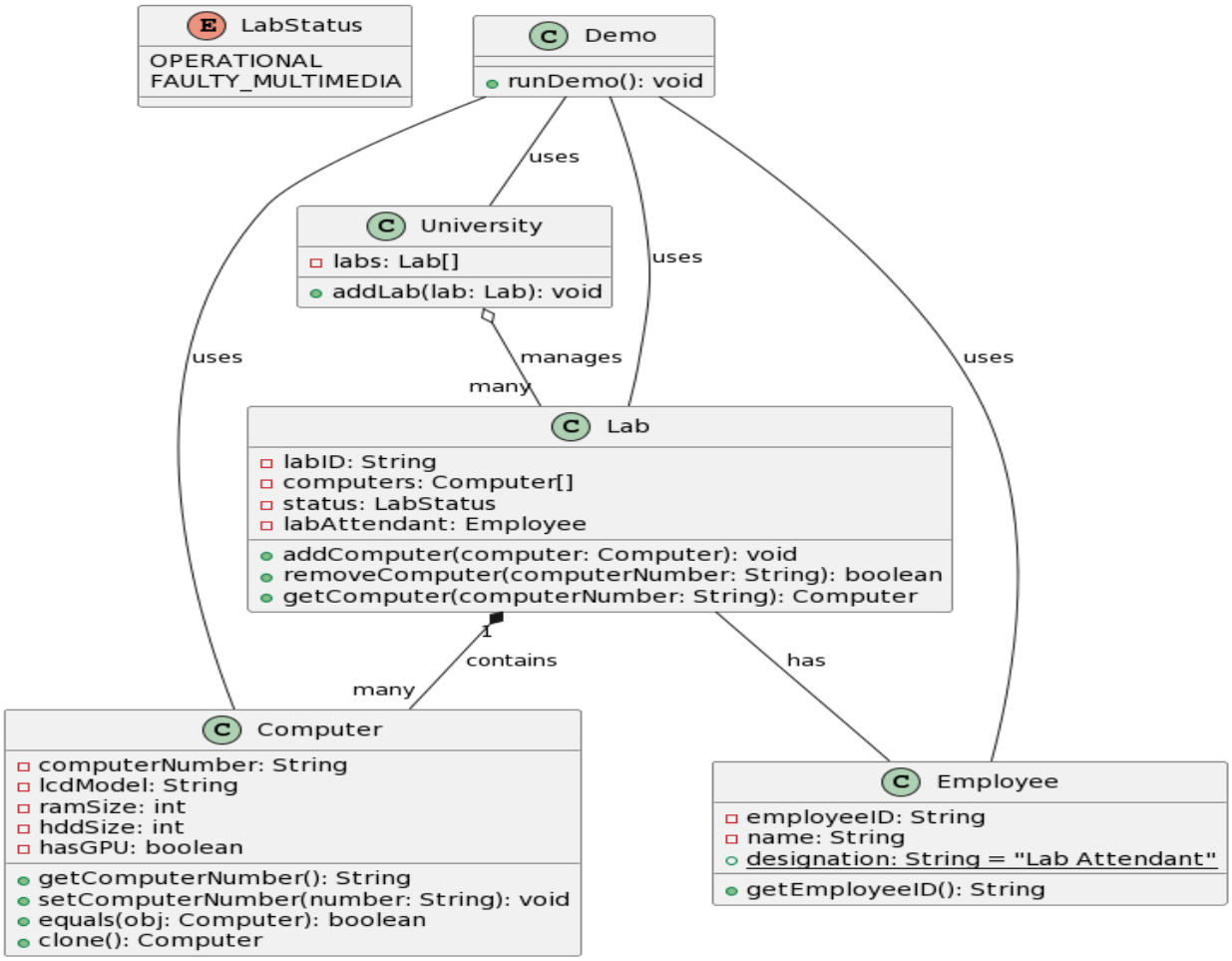
Lab Assignment 1

Date: 2-10-2023 **Maximum Marks:10**

Computer Lab Inventory Management System

The Computer Science Department of CUI maintains an inventory of its various computer labs. This inventory records all the systems and accessories available in the labs. Each department, such as Pharmacy, might have a different number of labs. Each lab houses several computer systems, and all labs are equipped with multimedia, although some may exhibit faulty behavior. Every computer system has attributes like computer number, LCD model, RAM size in GBs, hard disk size in GBs, and whether or not a GPU is available. A Lab Attendant, an employee of the university, oversees every lab.



Your Task:

You must develop a Java application that effectively manages this inventory while employing object-oriented principles you've learned so far.

Specifications:

1. **Class Computer**:
    - Attributes:

        - **computerNumber** (String): Unique identifier for each computer.
        - **lcdModel** (String): Model of the LCD.
        - **ramSize** (int): RAM size in GBs.
        - **hddSize** (int): Hard Disk size in GBs.
        - **hasGPU** (boolean): True if GPU is available, False otherwise.
    - Constructors:
        - Default constructor
        - Parameterized constructor
    - Methods:
        - Getter and Setter methods for all attributes.
        - **equals()**: To compare two computer objects.
        - Implement **clone()** to create a deep copy of the computer object.
2. **Enum LabStatus**:
    - Values: **OPERATIONAL**, **FAULTY_MULTIMEDIA**
3. **Class Lab**:
    - Attributes:
        - **labID** (String): Unique identifier for each lab.
        - **computers** (Array of **Computer** objects): Stores all computers in the lab.
        - **status** (LabStatus): Current status of the lab.
        - **labAttendant** (Employee): The attendant overseeing the lab.
    - Constructors:
        - Default constructor
        - Parameterized constructor
    - Methods:
        - Add, remove or fetch a computer.
        - Change lab status.
        - Set or get lab attendant details.
4. **Class Employee**:
    - Attributes:
        - **employeeID** (String): Unique identifier.
        - **name** (String): Name of the employee.
        - **designation** (String): Always set to "Lab Attendant" (This can be set using a **static** attribute).
    - Constructors:
        - Default constructor
        - Parameterized constructor
    - Methods:
        - Getter and Setter methods for all attributes.
5. **Class University**:
    - Attributes:
        - **labs** (Array of **Lab** objects): Stores all the labs of the computer science department.
    - Methods:
        - Add, remove or fetch a lab.

- Fetch details about a specific lab or computer.

Guidelines:

1. Use encapsulation: Ensure that all attributes of the classes are private and can be accessed/modified through public methods.
2. Use arrays to store multiple objects of **Computer** in the **Lab** class and multiple **Lab** objects in the **University** class.
3. Utilize both static and non-static attributes/methods where appropriate. For example, the **designation** in the **Employee** class can be a static attribute since it's common to all lab attendants.
4. Implement **equals()** method in the **Computer** class to check the equality of two computer objects based on their unique identifiers (**computerNumber**).
5. Implement the **clone()** method in the **Computer** class to create and return a deep copy of a computer object.

Your final application should allow the university to seamlessly manage its computer lab inventory, ensuring efficient tracking of systems and accessories.

Submission Instructions:
To ensure a seamless and standardized code submission process, please follow the guidelines outlined below when uploading your complete code to your personal GitHub repository. This will not only help maintain uniformity but also assist in easy navigation and review of your code. Create a new repository with a meaningful name related to our project, e.g., ComputerLabInventorySystem. Make sure you set the repository to be public so it can be reviewed.

==End==