

OBJECT ORIENTED SYSTEMS

Time: Three hours

Full Marks: 100

Different parts of the same question should be answered together.

CO1 [20]	<p>[1] Answer (a) and any one out of (b) and (c) from this block.</p> <p>(a) Implement an interface <i>Counter</i> in Java. The interface contains three methods called (i) <i>display()</i>, (ii) <i>incrementByOne()</i> and (iii) <i>decrementByOne()</i>.</p> <p>Create two classes called <i>IntegerCounter</i> and <i>EnumerationCounter</i> that implement <i>Counter</i>. <i>IntegerCounter</i> class contains an integer whose value is incremented or decremented or displayed. <i>EnumerationCounter</i> class contains an array of strings. When an <i>EnumerationCounter</i> object is incremented or decremented then the current index in the array is increased or decreased by one till the index reaches its maximum or minimum value. If the index reaches maximum value, an increment in the index will set the index to zero. If the index reaches zero, a decrement in the index will set the index to its maximum value.</p> <p>Implement the classes <i>IntegerCounter</i>, <i>EnumerationCounter</i> in Java. [10]</p> <p>(b) Answer the following questions.</p> <p>(i) What are checked and unchecked exceptions? Give examples.</p> <p>(ii) What is the base class of all classes in Java? Write 3 methods of that class.</p> <p>(iii) Why do we (programmers/designers) write interfaces?</p> <p>(iv) Which methods in the base class are not overridden? [(2+2)+(1+2)+1+2]</p> <p>(c) Answer the following questions.</p> <p>(i) Consider the following two class definitions.</p> <pre> class Outer { int x; class Inner extends Unrelated{ void inc () { x++; } } } class Outer { void print() { } void print (intval) { } class Inner { void print() { } void show() { print(); Outer.this.print(); print(1); } } } </pre> <p>Identify what is wrong in those classes or state that the definition is correct. [3+2]</p> <p>(ii) What are the uses of <i>final</i> keyword? What are the differences between <i>String</i> class and <i>StringBuffer</i> class? Give two examples of Marker interfaces. [2+2+1]</p>
CO2 [20]	<p>[2]</p> <p>All elements of a huge matrix are to be added. To expedite processing, 4 threads are created. Suppose that</p>

	<p>the matrix is global to all these threads. Each thread takes a column of that matrix, adds the elements of column and the sum is accumulated in a global variable. When a thread completes processing of one column then it takes up another column which has not yet been processed and is not being processed currently by other threads.</p> <p>Write the necessary Java classes for this problem.</p>
<p>CO3 [20]</p>	<p>[3] Answer any one out of (a) and (b) from this block.</p> <p>(a) Implement a Java program that accepts the name of a Java class, and constructs a Java interface for given class using Reflection API. The interface must include: Method headers Return value data types Parameter data types Constants (attributes declared as final)</p> <p>(b)</p> <p>(i) Define the method setValue (Object obj, String "FieldName", int value) so that the field name "FieldName" of the object "obj" is set to the integer "value".</p> <p>(ii) Define the method createObject (String className, Object [] args) to create an object of the class "className" using a constructor having the same number of arguments as "args" array. Use the "args" array the arguments for the constructor.</p> <p style="text-align: right;">[10+]</p>
<p>CO4 [20]</p>	<p>[4] Answer question (c) and any one question out of (a) and (b) from this block.</p> <p>(a) Draw the Sequence Diagram corresponding to the following code.</p> <pre> class Customer { public void iWannaEat() { DinnerNowSystem dns = new DinnerNowSystem(); Restaurant r = new Restaurant(); Bank b = new Bank(); while (...) dns.addOrderItem(); dns.confirmOrder(); dns.sendorder(); dns.paymentDetail(); Food f = r.receiveFood(); } } class DinnerNowSystem() { Bank b; Restaurant r; void addOrderItem(); void confirmOrder() { r.sendOrder(); } void paymentDetail() { if (b.processPayment()) r.confirmOrder() } } </pre>

(b) Draw a class diagram to model the relationship among classes from the following description.
 A hotel has a number of rooms that can be rent by guests. There are also a number of bathrooms, which are either connected to a specific room or are used to service multiple rooms on the floor. The rooms are classified into three types: single rooms, double rooms and family rooms. Each single room can only be rent to at most one guest. Each double room can be rent to at most two guests. Each family room can be rent to a family of up to two adults and two children

[10]

(c) Read the following passage and draw the statechart diagram that describes the lifetime of a collectible item from the moment it is ordered by the library until it is removed from the library. For each state, specify entry, exit, and do actions. For each transition, specify relevant events, conditions and actions.

- The database of a library holds information on collectible items such as books, videos and CDs available for borrowing.
- Each user can borrow an item for two weeks. If the item is not returned by that time, the user pays a fine.
- When a user selects an item to be borrowed, the library clerk updates the database, recording userID, collection#, and date of borrowing;
- When a user returns an item, the library clerk updates the database, recording the date when the item is returned; if the item is overdue, the clerk also collects a fine and the clerk also records (in the database) the fine collected, if any; in addition, the clerk checks if the returned item is damaged, and if so, records this information (in the database, again).
- To add new material to the collection, the library orders it from XYZ Co., a wholesale supplier. When it arrives, it is catalogued (i.e., an entry is added to the database) and it is made available in the library.
- Damaged and unused material is removed from the collection. This is done once every six months. The process consists of selecting all items that are damaged, or haven't been borrowed in the last year, and physically removing them from the collection.

[10]

[5] Answer any one out of (a) and (b) from this block.

(a)

(i)

```

public class DataStructure {
    private final static int SIZE = 15;
    private int[] arrayOfInts = new int[SIZE];

    public DataStructure() {
        for (int i = 0; i < SIZE; i++) {
            arrayOfInts[i] = i;
        }
    }

    public void printEven() {
        DataStructureIterator iterator = this.newEvenIterator();
        while (iterator.hasNext()) {
            System.out.print(iterator.next() + " ");
        }
    }

    private class EvenIterator extends DataStructureIterator {
        public boolean hasNext() { // Write Code Here }
        public Integer next() { // Write Code Here }
    }

    public static void main(String s[]) {
        DataStructure ds = new DataStructure();
        ds.printEven();
    }
}
    
```

[10]

- (1) Complete the definition of the `DataStructure` class.
- (2) Define a method named `print(DataStructureIterator iterator)`. Invoke this method with an instance of the class `EvenIterator` so that it performs the same function as the method `printEven`.
- (3) Invoke the method `print(DataStructureIterator iterator)` so that it prints elements that have an odd index value. Use an anonymous class as the method's argument instead of an instance of the interface `DataStructureIterator`.

(ii) Show the structure of a visitor pattern using class diagram and sequence diagram.

[14+(3+3)]

(b)

(i) Define a class for a node in a binary tree. Implement a class `TreeLevelIterator` that has 3 methods: `hasNext()`, `next()`, and `currentNode()`. The iterator class traverses the tree level wise.

(ii) Show the structure of a visitor pattern using class diagram and sequence diagram.

[14+(3+3)]