



MAY 2022: END SEMESTER ASSESSMENT (ESA) B TECH VI SEMESTER

UE19CS353 – OBJECT ORIENTED ANALYSIS AND DESIGN WITH JAVA

Time: 3 Hrs	Answer All Questions	Max Marks: 100
1	a) State any six differences between abstract class and interface?	6
	b) State the reason why the given code prints output as false. Also change the code such that the output in the main function is true. <pre>public class MainClass { public static void main(String[] args) { ClassOne objOne=new ClassOne(10,20); ClassOne objTwo=new ClassOne(10,20); System.out.println(objOne.equals(objTwo)); //prints false } } class ClassOne{ int a; int b; ClassOne(int a,int b){ this.a=a; this.b=b; } }</pre>	4
	c) Write code to add the following functionality expected by the MainClass. The Teacher object can be created in two ways. The Anchor is also a Teacher. In addition, the Anchor has an array/list of Teachers who are his/her co-teachers. <pre>class Teacher{ private String name; private String subject; //Write code here }</pre> <pre>// Write code here</pre> <pre>class MainClass{ //Do not change this method public static void main(String[] args) { Teacher t1 = new Teacher("Teacher Name"); Teacher t2 = new Teacher("Teacher Name", "Teacher Subject"); Anchor a1 = new Anchor("Teacher Name", "Teacher Subject"); a1.addTeacher(t1); a1.addTeacher(t2); a1.showDetails(); // shows details of Anchor as well as all his/her co-teachers } }</pre> Note: only write the code expected. Do not repeat the given code.	6
	d) What are the two types of methods/attributes that can be defined in Java? What is the keyword used to differentiate between the two? Write code snippet for the same.	4

2	a)	<p>Given below is the class Employee with its properties and methods. The class is serializable. Write code to read an object of Employee from the file "e1.txt" and update the salary by 10% and write back the object to the same file.</p> <pre>import java.io.*; import java.io.Serializable; public class Employee implements Serializable{ Double salary; Int age; String name; public Employee(Long salary, Int age ,String name) { this.salary = salary; this.age = age; this.name = name; } public updateSalary(Float percent){ this.salary = (1.0+percent)*this.salary; } } class Persist{ public static void main(String args[]){ try{ // WRITE CODE }catch(Exception e){System.out.println(e);} } }</pre> <p>Note: Write only the code expected. Do not repeat the given code.</p>	6
	b)	<p>Draw a Use Case diagram of an Airline Seat Reservation System.</p> <p>In the system, a Passenger logs to the system and checks for flights available. The passenger can book a ticket and optionally choose a seat of their choice and accordingly make a card payment for it. The bank validates the payment for the system. Passengers can also cancel their ticket. The system admin processes the cancellation as well as update flight when required.</p>	4
	c)	<p>Draw the Class modeling diagram to model a system for management of flights and pilots</p> <ul style="list-style-type: none"> • An airline operates flights. • Each airline has an ID and Name. • Each flight has an ID a departure airport and an arrival airport • An airport has a unique identifier. • Each flight has at least a pilot and a co-pilot, and it uses an aircraft of a certain type. • A flight also has a departure time and an arrival time. • An airline owns a set of aircrafts of different types. • An aircraft can be in a working state or it can be under repair. • In a particular moment an aircraft can be landed or airborne. • A company has a set of pilots: each pilot has an experience level: 1 to 3 • A type of aircraft may need a particular number of pilots, with a different role (e.g.: captain, co-pilot, navigator): there must be at least one captain and one co-pilot, and a captain must have a level 3. 	6
	d)	<p>What are the different elements in a Deployment Diagram? Write the notations for these in an example diagram.</p>	4

3

<p>b)</p>	<p>What SOLID design principle is violated in the given design? How can this be modified to make it a better design? Draw class diagram and code snippet of your design.</p> <div data-bbox="599 216 1102 550"> <pre> classDiagram class Mammal { <<interface>> +walk() +fly() +swim() } class Human class Bat class Whale Human .. > Mammal Bat .. > Mammal Whale .. > Mammal </pre> </div>	<p>6</p>
<p>c)</p>	<p>Describe the Singleton creational pattern with a suitable code snippet.</p>	<p>4</p>
<p>d)</p>	<p>Fill in the blanks:</p> <ol style="list-style-type: none"> Violating Dependency Inversion Principle will result in a code with _____ coupling. Liskov Substitution Principle states that the designer/implementer should be able to replace objects of _____ with the objects of _____. _____ Principle states that the design should be extensible however modifying existing design should be avoided. The pattern that hides the complexity related to creation of objects from the client is _____. 	<p>4</p>
<p>5</p>	<p>a) Implement the following use case using a proxy pattern. Write both the class diagram and the code snippet. State which type of proxy pattern is suitable for the given use case.</p> <p>Use Case:</p> <p>If an entire library of books, with all the details about each book, is loaded from a database, it will consume a lot of RAM, and it is very likely that the user will need to issue or return <i>only</i> one book. The solution, that uses the proxy design pattern, displays only the <i>name</i>, <i>author</i>, and <i>availability</i> of the books when a list of them is displayed. When a book is selected, all of the remaining details will be fetched from the database and operations like issue/return (update availability) can be performed.</p> <p>Note: do not write the entire code; write only code snippets/pseudo code.</p> <p>b) For a case study shown below, identify the appropriate structural design pattern. Write the class structure of the identified pattern.</p> <div data-bbox="519 1407 1143 1722"> <pre> graph LR SDP[Stock Data Provider] -- XML --> App subgraph Application CC[Core Classes] end App -- XML --> CC CC -- XML --> XJ[XML to JSON] XJ -- JSON --> AL[Analytics Library] </pre> </div>	<p>6</p>
<p>c)</p>	<p>Describe the Chain of responsibilities pattern with an example.</p>	<p>4</p>
<p>d)</p>	<p>Mention symptoms and typical causes for Vendor-Lock-In design antipattern.</p>	<p>6</p>