

Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 145 Question Id : 640653737420 Question Type : SA Calculator : None
Response Time : N.A Think Time : N.A Minimum Instruction Time : 0
Correct Marks : 4

Question Label : Short Answer Question

Consider an encoder-decoder pair used for dimensionality reduction, where the encoder function is denoted as $f(x_1, x_2, x_3) = (x_1 + x_2 - x_3, x_1 + x_3)$ and the decoder function is denoted as $g(u) = (u_1, u_2, u_1 + u_2)$

Compute the reconstruction error $\left(R(f, g) = \frac{1}{n} \sum_{i=1}^n \|X_i - g(f(X_i))\|^2\right)$ for the following dataset:

$$X_1 = [1, 2, 3], \quad X_2 = [0, 2, 2], \quad X_3 = [1, 0, -1]$$

Enter the answer correct to two decimal places.

Response Type : Numeric
Evaluation Required For SA : Yes
Show Word Count : Yes

Answers Type : Range
Text Areas : PlainText

Possible Answers :

3.12 to 3.20

Java

Section Id :	64065351358
Section Number :	10
Section type :	Online
Mandatory or Optional :	Mandatory
Number of Questions :	16
Number of Questions to be attempted :	16
Section Marks :	100

Display Number Panel :	Yes
Section Negative Marks :	0
Group All Questions :	No
Enable Mark as Answered Mark for Review and Clear Response :	Yes
Maximum Instruction Time :	0
Sub-Section Number :	1
Sub-Section Id :	640653107637
Question Shuffling Allowed :	No
Is Section Default? :	null

Question Number : 146 Question Id : 640653737424 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 0

Question Label : Multiple Choice Question

THIS IS QUESTION PAPER FOR THE SUBJECT "DIPLOMA LEVEL : PROGRAMMING CONCEPTS USING JAVA (COMPUTER BASED EXAM)"

ARE YOU SURE YOU HAVE TO WRITE EXAM FOR THIS SUBJECT?
CROSS CHECK YOUR HALL TICKET TO CONFIRM THE SUBJECTS TO BE WRITTEN.

(IF IT IS NOT THE CORRECT SUBJECT, PLS CHECK THE SECTION AT THE TOP FOR THE SUBJECTS REGISTERED BY YOU)

Options :

6406532468009.  YES

6406532468010.  NO

Sub-Section Number :	2
Sub-Section Id :	640653107638
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 147 Question Id : 640653737425 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 6 Max. Selectable Options : 0

Question Label : Multiple Select Question

Which of the following statements is/are true about activation records?

Options :

6406532468011. ✖ All the variables on the stack (in any activation record) are said to be in scope.

6406532468012. ✖ The lifetime of a variable is the duration that it spends in the topmost activation record in the stack.

6406532468013. ✔ Every activation record has a control link that points to start of previous record.

6406532468014. ✔ Every activation record has a return value link that points to where to store return value.

Question Number : 148 Question Id : 640653737426 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 6 Max. Selectable Options : 0

Question Label : Multiple Select Question

Consider the statements given below.

Statement 1 : Type 2DShape has method : `getArea()`.

Statement 2 : Type 3DShape has two methods : `getArea()` and `getVolume()`.

Choose the correct option(s) regarding subtyping and inheritance with respect to type 2DShape and type 3DShape.

Options :

6406532468015. ✔ 3DShape can be a subtype of 2DShape.

6406532468016. ✖ 2DShape can be a subtype of 3DShape.

6406532468017. ✓ 3DShape can be a subclass of 2DShape.

6406532468018. ✗ 2DShape can be a subclass of 3DShape.

Sub-Section Number :	3
Sub-Section Id :	640653107639
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 149 Question Id : 640653737430 Question Type : MCQ Is Question
Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction
Time : 0
Correct Marks : 6
Question Label : Multiple Choice Question

Consider the code given below.

```
class Athlete {
    public void perform() {
        System.out.println("Athlete performs");
    }
    public void perform(String activity) {
        System.out.println("Athlete performs " + activity);
    }
}
class Gymnast extends Athlete {
    public void perform(String move) {
        System.out.println("Gymnast performs " + move);
    }
}
public class TestAthlete {
    public static void main(String[] args) {
        Athlete a = new Gymnast(); // LINE 1
        a.perform();
        a.perform("backflip"); // LINE 2
    }
}
```

Choose the correct option.

Options :

6406532468031. ✖ LINE 1 generates compilation error because a variable of type `Athlete` cannot refer to an object of type `Gymnast`.

6406532468032. ✖ This code generates the below output followed by runtime Error at LINE 2 because there is ambiguity in which `perform( )` method is being invoked.
Athlete performs

6406532468033. ✖ This code generates the output:
Athlete performs
Athlete performs backflip

6406532468034. ✔ This code generates the output:
Athlete performs
Gymnast performs backflip

Question Number : 150 Question Id : 640653737434 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 6

Question Label : Multiple Choice Question

Consider the Java code given below.

```
class Vehicle {
    public final void startEngine() {
        System.out.println("Starting the engine of a vehicle");
    }
}
class Car extends Vehicle {
    public final void startEngine() {
        System.out.println("Starting the engine of a car");
    }
}
public class TestVehicles {
    public static void main(String[] args) {
        Vehicle vehicle = new Car();
        vehicle.startEngine();
    }
}
```

Choose the correct option regarding the given code.

Options :

This code generates output:

Starting the engine of a vehicle
Starting the engine of a car

6406532468047. ✖

This code generates output:

Starting the engine of a vehicle

6406532468048. ✖

This code generates output:

Starting the engine of a car

6406532468049. ✖

6406532468050. ✓ This code generates a compilation error because the method `startEngine` cannot be overridden

Question Number : 151 Question Id : 640653737436 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 6

Question Label : Multiple Choice Question

Consider the Java code given below.

```
interface Readable {
    default void read() {
        System.out.println("reads");
    }
}
class Book implements Readable { //LINE 1
}
class Newspaper implements Readable {
    public void read() {
        System.out.println("reads news");
    }
}
public class Test {
    public static void main(String[] args) {
        Readable r1 = new Book();
        r1.read(); // LINE 2
        Readable r2 = new Newspaper();
        r2.read();
    }
}
```

Choose the correct option.

Options :

6406532468055. ✖ LINE 1 generates compilation error because class `Book` is not abstract

This code generates the output:

reads

6406532468056. ✓ reads news

This code generates the output:

reads

6406532468057. ✖ reads

LINE 2 generates runtime error because method read() is not defined in class Book

6406532468058. ✖

Question Number : 152 Question Id : 640653737437 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 6

Question Label : Multiple Choice Question

Consider the Java code given below.

```
interface Playable {
    default void play() {
        System.out.println("Plays");
    }
}
interface Pauseable {
    default void pause() {
        System.out.println("Pauses");
    }
}
class MediaPlayer implements Playable, Pauseable{ //LINE 1
    public void play(){
        System.out.println("Plays media player");
    }
}
public class Test {
    public static void main(String[] args) {
        Playable p1 = new MediaPlayer();
        p1.play();
        p1.pause(); //LINE 2
    }
}
```

Choose the correct option.

Options :

6406532468059. ✖ LINE 1 generates compilation error because the class MediaPlayer is not declared as abstract.

6406532468060. ✔ LINE 2 generates compilation error because p1 of type Playable cannot invoke method pause().

6406532468061. ✖ This code generates the output:
Plays media player
Pauses

6406532468062. ✖ This code generates the output:
Plays
Pauses

Question Number : 153 Question Id : 640653737438 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0
Correct Marks : 6
Question Label : Multiple Choice Question

Consider the code given below.

```
interface PaymentMethod {
    void processPayment();
}

class Payment {
    public CreditCard getPaymentMethod1() {
        return new CreditCard();
    }
    public PayPal getPaymentMethod2() {
        return new PayPal();
    }
    private class CreditCard implements PaymentMethod {
        public void processPayment() {
            System.out.println("Processing credit card payment");
        }
    }
    private class PayPal extends CreditCard {
        public void processPayment() {
            System.out.println("PayPal payment started");
        }
    }
}

public class PrivateTest {
    public static void main(String[] args) {
        Payment p = new Payment();
        // CODE BLOCK
        obj1.processPayment();
        obj2.processPayment();
    }
}
```

Choose the correct option to fill in place of CODE BLOCK so that the output is:

Processing credit card payment
PayPal payment started

Options :

CreditCard obj1 = new CreditCard();
6406532468063. ✖ PayPal obj2 = new PayPal();

PaymentMethod obj1 = p.getPaymentMethod1();
6406532468064. ✔ PaymentMethod obj2 = p.getPaymentMethod2();

6406532468065. ✖ `CreditCard obj1 = p.getPaymentMethod1();
PayPal obj2 = p.getPaymentMethod2();`

6406532468066. ✖ `PaymentMethod obj1 = new CreditCard();
PaymentMethod obj2 =new PayPal();`

Sub-Section Number :	4
Sub-Section Id :	640653107640
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 154 Question Id : 640653737427 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 7

Question Label : Multiple Choice Question

Consider the Java code given below.

```
public class PatternSwitchExample {
    public static void displayPattern(int value) {
        switch (value) {
            case 1:
                System.out.print("A ");
                break;
            case 2:
                System.out.print("B ");
                break;
            case 3:
                System.out.print("C ");
                break;
            default:
                System.out.print("X ");
                break;
        }
    }
    public static void main(String[] args) {
        for (int i = 3; i >= 1; i--) {
            for (int j = 1; j <= i; j++) {
                displayPattern(i);
            }
            System.out.println();
        }
    }
}
```

What will the output be?

Options :

C C C

B B B

6406532468019. ✖ A A A

A

B B

6406532468020. ✖ C C C

C C C

B B

6406532468021. ✔ A

X X X

C B

6406532468022. ✖ A

Question Number : 155 Question Id : 640653737428 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 7

Question Label : Multiple Choice Question

Consider the Java code given below.

```
class Student {
    private String name;
    private int age;
    public Student(String n, int a) {
        name = n;
        age = a;
    }
    public Student(Student s) {
        this.name = s.name;
        this.age = s.age;
    }
    public void setName(String n) {
        name = n;
    }
    public void setAge(int a) {
        age = a;
    }
    public int getAge() {
        return age;
    }
    public String getName() {
        return name;
    }
}

public class StudentDemo {
    public static void main(String[] args) {
        Student s1 = new Student("Alice", 20);
        Student s2 = s1;
        s1.setAge(25);
        s1.setName("Emily");
        System.out.println("s2:name: " + s2.getName() + ", age: " + s2.getAge());
    }
}
```

What will the output be?

Options :

6406532468023. ✓ s2:name: Emily, age: 25

6406532468024. ✗ s2:name: Emily, age: 20

6406532468025. ✖ s2:name: Alice, age: 25

6406532468026. ✖ s2:name: Alice, age: 20

Question Number : 156 Question Id : 640653737429 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 7

Question Label : Multiple Choice Question

Consider the Java code given below.

```
class Car {
    private String model;
    private int year;
    public Car(String m, int y) {
        model = m;
        year = y;
    }
    ***-----***
    * CODE SEGMENT *
    ***-----***
}
public class CarTest {
    public static void main(String[] args) {
        Car car1 = new Car("Toyota", 2022);
        Car car2 = new Car("Ford", 2020);
        System.out.println(car1 + "\n" + car2);
    }
}
```

Choose the correct option to fill in the CODE SEGMENT so that the output is:

Toyota : 2022

Ford : 2020

Options :

```
public String toString(){
    return model + " : " + year;
```

6406532468027. ✔ }

```
public String toString(Object ob){  
    return ob.model + " : " + ob.year;  
}
```

6406532468028. ✖

```
String toString(){  
    return model + " : " + year;  
}
```

6406532468029. ✖

```
void String toString(Object ob){  
    System.out.println(ob.model + " : " + ob.year);  
}
```

6406532468030. ✖

Question Number : 157 Question Id : 640653737431 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 7

Question Label : Multiple Choice Question

Consider the following code:

```
class Animal {
    public void makeSound() {
        System.out.println("Animal makes a sound");
    }
}
class Dog extends Animal {
    public void makeSound(String s) {
        makeSound();
        System.out.println("Dog " + s);
    }
}
public class TestAnimals {
    public static void main(String[] args) {
        Animal myPet = new Dog();
        //CODE SEGMENT
    }
}
```

Choose the correct option to fill in the CODE SEGMENT so that the output is:

Animal makes a sound

Dog barks

Options :

6406532468035. ✖ myPet.makeSound();

6406532468036. ✖ myPet.makeSound("barks");

6406532468037. ✖ Animal a = (Animal)myPet;
a.makeSound("barks");

6406532468038. ✔ Dog d = (Dog)myPet;
d.makeSound("barks");

Question Number : 158 Question Id : 640653737433 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 7

Question Label : Multiple Choice Question

Consider the code given below that checks whether two Persons are the same. Method equals is overridden to compare two Person objects as follows. If two Persons have the same name and age, then they are the same. Based on the given information, answer the question that follows.

```
class Person {
    private String name;
    private int age;

    // Constructor to initialize instance variables

    public String toString() {
        return name + " (" + age + " years old)";
    }
    public boolean equals(Object obj) {
        //CODE BLOCK
    }
}

public class TestPerson {
    public static void main(String[] args) {
        Person person1 = new Person("Alice", 25);
        Person person2 = new Person("Alice", 25);
        if (person1.equals(person2)){
            System.out.println(person1 + " and " + person2 + " are the same");
        }
        else{
            System.out.println(person1 + " and " + person2 + " are different");
        }
    }
}
```

Choose the correct option to fill in place of CODE BLOCK so that the output is:
Alice (25 years old) and Alice (25 years old) are the same

Options :

```
if(obj instanceof Person) {
    if(this.age.equals(obj.age) && this.name.equals(obj.name))
        return true;
}
return false;
```

6406532468043. ✖

6406532468044. ✖

```
if(this.age.equals(obj.age) && this.name.equals(obj.name))  
    return true;  
return false
```

```
if(obj instanceof Person){  
    Person otherPerson = (Person) obj;  
    if(this.age == otherPerson.age && this.name.equals(otherPerson.name)){  
        return true;  
    }  
}
```

6406532468045. ✓ return false;

```
if(obj instanceof Person) {  
    Person otherPerson = obj;  
    if(this.age == otherPerson.age && this.name.equals(otherPerson.name))  
        return true;  
}  
return false;
```

6406532468046. ✗

Question Number : 159 Question Id : 640653737435 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction

Time : 0

Correct Marks : 7

Question Label : Multiple Choice Question

Consider the Java code given below.

```
1 interface Switchable {
2     public void turnOn();
3     public void turnOff();
4 }
5 interface SpeedAdjustable {
6     public void adjustSpeed();
7     default void displaySpeed() {
8         System.out.println("Speed is adjustable.");
9     }
10 }
11 class Fan implements Switchable,SpeedAdjustable {
12     public void turnOn() {
13         System.out.println("Fan is turned on.");
14     }
15     public void turnOff() {
16         System.out.println("Fan is turned off.");
17     }
18 }
```

Choose the correct option regarding the above code.

Options :

6406532468051. ✖ LINE 7 generates compilation error because the method displaySpeed is not abstract.

6406532468052. ✖ LINE 11 generates compilation error because class Fan cannot implement two interfaces.

6406532468053. ✔ LINE 11 generates compilation error because class Fan is not declared as abstract.

6406532468054. ✖ LINE 2, LINE 3 & LINE 6 generate compilation errors because the methods turnOn, turnOff and adjustSpeed are not abstract.

Sub-Section Number :

5

Sub-Section Id :

640653107641

Question Shuffling Allowed :

Yes

Is Section Default? :

null

Question Number : 160 Question Id : 640653737432 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 8

Question Label : Multiple Choice Question

Consider the code given below.

```
class Shape {
    public void draw() {
        System.out.println("Drawing a shape");
    }
}
class Rectangle extends Shape {
    public void draw() {
        System.out.println("Drawing a rectangle");
    }
}
class Square extends Rectangle {
    public void draw() {
        System.out.println("Drawing a square");
    }
    public void color() {
        System.out.println("Square is colorful");
    }
}
public class TestShapes {
    public static void main(String[] args) {
        Shape s = new Rectangle();
        Rectangle r = new Square(); // LINE 1
        s.draw();
        r.color(); // LINE 2
    }
}
```

Choose the correct option.

Options :

LINE 1 generates compilation error because a variable of type Rectangle cannot refer to an object of type Square.

6406532468039. ✖

6406532468040.

This code generates the output:

Drawing a shape

✖ Square is colorful

LINE 2 generates compilation error because the method color() is not defined in class Rectangle.

6406532468041. ✔

This code generates the output:

Drawing a rectangle

Square is colorful

6406532468042. ✖

Question Number : 161 Question Id : 640653737439 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 8

Question Label : Multiple Choice Question

Consider the Java code given below.

```
interface Discountable {
    public default void isDiscountable() {
        System.out.println("Item eligible for discount");
    }
}

class Product {
    double price;
    public Product(double price) {
        this.price = price;
    }
    public void isDiscountable() {
        if (price <= 500.0) {
            System.out.println("Item not eligible for discount");
        }
        else {
            System.out.println("Discount granted");
        }
    }
}

class SaleItem extends Product implements Discountable {
    public SaleItem(double price) {
        super(price);
    }
}

public class TestClass {
    public static void main(String[] args) {
        Discountable i1 = new SaleItem(600.0);
        i1.isDiscountable();
        Discountable i2 = new SaleItem(300.0);
        i2.isDiscountable();
    }
}
```

Choose the correct option.

Options :

6406532468067. ✖ This program generates no output.

This code generates the output:

Item eligible for discount

6406532468068. ✖ Item not eligible for discount

This code generates the output:
Discount granted
6406532468069. ✓ Item not eligible for discount

This code generates the output:
Item not eligible for discount
Discount granted
6406532468070. ✖

AppDev2

Section Id :	64065351359
Section Number :	11
Section type :	Online
Mandatory or Optional :	Mandatory
Number of Questions :	17
Number of Questions to be attempted :	17
Section Marks :	50
Display Number Panel :	Yes
Section Negative Marks :	0
Group All Questions :	No
Enable Mark as Answered Mark for Review and Clear Response :	Yes
Maximum Instruction Time :	0
Sub-Section Number :	1
Sub-Section Id :	640653107642
Question Shuffling Allowed :	No
Is Section Default? :	null

Question Number : 162 Question Id : 640653737440 Question Type : MCQ Is Question