

BIRLA INSTITUTE OF TECHNOLOGY, MESRA, RANCHI
(MID SEMESTER EXAMINATION MO/2024)

CLASS: BTECH
BRANCH: CSE/AIML

SEMESTER : III/ADD
SESSION : MO/2024

SUBJECT: CS233 OBJECT ORIENTED PROGRAMMING & DESIGN PATTERN
TIME: 02 Hours

FULL MARKS: 25

INSTRUCTIONS:

1. The question paper contains 5 questions each of 5 marks and total 25 marks.
2. Attempt all questions.
3. The missing data, if any, may be assumed suitably.
4. Tables/Data handbook/Graph paper etc., if applicable, will be supplied to the candidates

-
- | | | CO | BL |
|---|--|-----|--------|
| Q.1(a) | How did java become a platform-independent language? | [2] | 1 2 |
| Q.1(b) | Consider the following scenarios and justify whether each of the scenarios is an example of abstraction or encapsulation properties of object-oriented programming. | [3] | 1 2 |
| <p>Scenario 1: You have a class <i>BankAccount</i> that provides methods like <i>deposit()</i> and <i>withdraw()</i> but hides the internal details of how the balance is stored or calculated. The user interacts only with the public methods and does not know how the calculations are performed.</p> <p>Scenario 2: A class <i>Car</i> has private variables like <i>speed</i> and <i>engineTemperature</i> and provides public methods <i>accelerate()</i> and <i>coolEngine()</i>. The internal state of the car's components is hidden and modified only through these methods.</p> <p>Scenario 3: An <i>Animal</i> class provides a method <i>makeSound()</i> implemented differently by subclasses like <i>Dog</i> and <i>Cat</i>. The concept of making a sound is defined at a high level, but the specific details are left to the subclasses.</p> | | | |
| Q.2(a) | Correct the errors (if any) and predict the output of the following code segment: class Test { public static void main(String args[]) { byte b; int i = 257; double d = 323.142; System.out.println("Conversion of int to byte."); b = (byte) i; System.out.println("i = " + i + " b = " + b); System.out.println("\nConversion of double to byte."); b = (byte) d; System.out.println("d = " + d + " b = " + b); } } | [2] | 2 4 |
| Q.2(b) | (i) Differentiate between local variable and instance variable. (ii) Differentiate between static and non-static method. (iii) Differentiate between 'Integer' and 'int' in context of java? | [3] | 2 4 |
| Q.3(a) | Can we overload Java main method? Justify your answer with an example. | [2] | 2 2 |
| Q.3(b) | What do you understand about constructors? Describe the types of constructors with an example? What is the behavior of constructors with keyword static? | [3] | 2 1, 4 |

PTO

Q.4(a)

What is the output of the following program? Explain your answer. Assume both class A and Main are in same packages.

[2] 2 3

| | |
|---|---|
| <pre> class A{ public int val; public A(int v){ this.val = v; } } </pre> | <pre> public class Main{ public static A modifyVal(A a){ a.val = 30; return a; } public static void main(String args[]){ A a1 = new A(10); A a2 = a1; a2.val = 20; A a3 = modifyVal(a2); System.out.println(a1.val); System.out.println(a2.val); System.out.println(a3.val); } } </pre> |
|---|---|

Q.4(b)

Explain three usages of 'this' keyword with example? Can we use 'this' with a static method? If yes then provide a suitable example, if no then justify your answer.

[2+1] 2 1, 4

Q.5(a)

Illustrate the difference between standard and enhanced for statement (for-each loop) with ArrayList.

[2] 2 3

Q.5(b)

Write a Java program to return true if an array can be split into two halves in such a way that the sum of left side of the splitting is equal to the sum of the right side.

[3] 2 3

Sample input 1: The given array is : 1 3 3 8 4 3 2 3 3

Sample output 1: True!! The array can be split in a position where the sum of both side are equal.

Note: Here, splitting position is in between 8 and 4. Sum of left side is 1+3+3+8=15 Sum of right side is 4+3+2+3+3 = 15

Sample input 2: The given array is : 1 4 1 8 4 3 5 3 3

Sample output 2: False!! The array cannot be split at any position where the sum of both side are equal.