U.S.N.										

B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute Affiliated to VTU

October 2024 Supplementary Examinations

Programme: B.E. **Semester: III**
Branch: Computer Science and Engineering **Duration: 3 hrs.**
Course Code: 23CS3PCOOJ **Max Marks: 100**
Course: Object Oriented Java Programming

Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

Important Note: Completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages. Revealing of identification, appeal to evaluator will be treated as malpractice.

		UNIT - I	*CO*	*PO*	Marks
1	a)	Demonstrate the uses of static keyword with an example program.	*CO1*	*PO1*	**6**

	b)	Analyze the errors in the given program. Write the corrected output. Underline the places where errors are corrected. Don't comment any complete statement(s).
		class Box {
		double width; double height; double depth;
		
		void volume ( )
		{
		System.out.print("Volume is ");
		System.out.println(width * height * depth);
		return (width * height * depth);
		} }
		
		class BoxDemo3
		{
		public static void main (String args[])
		{
		Box mybox1 = new Box( );
		Box mybox2;
		Box mybox1 = new Box (10,10,10);
		mybox1.width = 10; mybox1.height = 20; mybox1.depth = 15;
		mybox2.width = 3; mybox2.height = 6; mybox2.depth = 9;
		mybox1.volume( );
		mybox2.volume( );
		mybox3.volume( );
		}}
			CO2	*PO2*	**6**

	c)	Create a class Age with instance members years and months. Include methods to add two Age objects considering the fact that 12 months => 1 year and return back the resultant Age object.	*CO3*	*PO3*	**8**
		UNIT - II			
2	a)	Demonstrate generics with an example program.	*CO1*	*PO1*	**6**

		b)	Analyze the following program and write the expected output. Explain with proper justification. import java.lang.StringBuffer; public class Test { public static void main(String args[]) { String s1 = "java"; String s2 = "java"; System.out.println(s1.equals(s2)); System.out.println(s1 == s2); StringBuffer sb=new StringBuffer("Tomorrow"); System.out.println("length="+sb.length( )); System.out.println("capacity="+sb.capacity( )); }}	*CO2*	*PO2*	**6**
		c)	Create a class Employee with members: emp_id, emp_name, emp_salary, bonus. Include an abstract method cal_bonus. Create two subclasses Salesman with a member - sales and Developer with member- projects_completed. If the sales completed by the salesman is more than 10, bonus will be 5000 else bonus will be 1000. And for developer, if num of projects completed is more than 2, bonus will be 10000 else bonus will be 2000.	*CO3*	*PO3*	**8**
			UNIT - III			
	3	a)	Demonstrate finally keyword with an example program.	*CO1*	*PO1*	**6**
		b)	Analyze the given four program and identify the errors in each of the classes defined below. Justify your answer.	*CO2*	*PO2*	**6**

package p1;
public class Protection {
int n = 1; private int n_pri = 2;
protected int n_pro = 3;
public int n_pub = 4;
public Protection() {
System.out.println("base constructor");
}}
	package p1;
	public class Protection2 {
	Protection2 () {
	System.out.println(n+n_pri +n_pro+ n_pub);}}

package p1;
public class Protection1 extends Protection {
public Protection1(){
System.out.println(n+n_pri +n_pro+ n_pub);}}
	package p2;
	import p1.Protection;
	class Protection3 {
	Protection3 () {
	p1.Protection p = new p1.Protection();
	System.out.println(n+p.n_pri +p.n_pro+ n_pub);}}

		c)	Develop a java program to implement a stack that doubles in size when an overflow condition is encountered. Use Interfaces.	*CO3*	*PO3*	**8**
			OR			

	4	a)	Define a package. Tabulate the various levels of access protection available for packages and their implications.	*CO1*	*PO1*	**6**

		b)	Analyze the given program find the errors. Write the corrected program. Underline the places where errors are corrected and justify.
			
			interface MyInterface
			{
			void Method1( );
			void Method2( );
			}
			class Demo extend MyInterface
			{
			void method1( )
			{System.out.println("Implementation of method1");}
			void method2( )
			{System.out.println("Implementation of method2");}
			public static void main(String args[])
			{MyInterface obj=new MyInterface( );
			obj.method1( );
			}
			}
				CO2	*PO2*	**6**

		c)	Create a class BankAccount members- accountid and amount, methods deposit, withdraw and showbalance. Create a user defined Exception InsufficientFundsException that is thrown when the withdrawal amount is more than balance amount. Write BankDemo program to demonstrate invoking the deposit() and withdraw() methods of BankAccount.	*CO3*	*PO3*	**8**
			UNIT - IV			
	5	a)	Demonstrate FileOutputStream with an example program.	*CO1*	*PO1*	**6**

		b)	Modify the given program to get the output shown.
			Program:
			class A extends Thread { …… }
			class Main{
			public static void main(String args[]){
			A a1=new A();
			System.out.println("Name of thread 't':"+ t.getName( ));
			}
			}
			
			Expected Output:
			Name of thread 't': FirstThread
			New name of thread 't': NewThread
			Thread is running.
				CO2	*PO2*	**6**

		c)	Develop a java program to print the output in the following order only using threads. [BMS]	*CO3*	*PO3*	**8**

			[College] [of] [Engineering] Consider each of the words in output to be sent as a separate thread parameter.			
			OR			
	6	a)	Explain PushbackInputStream with an example program.	*CO1*	*PO1*	**6**

		b)	Analyze the program below and write the expected output.
			class PrintfDemo {
			public static void main(String args[]) {
			System.out.println("Here are some numeric values " +
			"in different formats.\n");
			System.out.printf("Various integer formats: ");
			System.out.printf("%d %(d %+d %05d\n", 3, -3, 3, 3);
			System.out.println();
			System.out.printf("Default floating point format: %f\n", 1234567.123);
			System.out.printf("Floating point with commas: %,f\n", 1234567.123)
			System.out.printf("Negative floating-point default: %,f\n", -1234567.123);
			System.out.printf("Negative floating point option: %,(f\n", -1234567.123);
			System.out.println();
			System.out.printf("Line up positive and negative values:\n");
			System.out.printf("% ,.2f\n% ,.2f\n", 1234567.123, -1234567.123);
			} }
				CO2	*PO2*	**6**

		c)	Develop a java program to create three threads using runnable Interface. Make all threads to execute for five iterations. Set the name of the three threads as "FIRST", "SECOND", "THIRD". Make the second thread to terminate for the 4th Iteration and last thread to sleep for two seconds after two iterations.	*CO3*	*PO3*	**8**
			UNIT - V			
	7	a)	Explain the AWT class hierarchy with a neat diagram.	*CO1*	*PO1*	**6**
		b)	Develop a Java program to draw a polygon of five sides and arc from 12 'o' clock to 6 'o' clock.	*CO3*	*PO3*	**6**
		c)	Develop a Java program that changes the background and foreground colors (random colors of your choice) on the press of the left mouse button and changes to default colors when the mouse is released.	*CO3*	*PO3*	**8**

****