U.S.N.										

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

Autonomous Institute Affiliated to VTU

September / October 2023 Supplementary Examinations

Programme: B.E.	**Semester: III**
Branch: Computer Science And Engineering	**Duration: 3 hrs.**
Course Code: 19CS3PCOOJ	**Max Marks: 100**
Course: Object Oriented Java Programming	**Date: 20.09.2023**

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

UNIT - I

1 a) Discuss on how Java is Robust and Architecture-Neutral **05**

b) Develop a menu driven Java program that accepts two numbers and an operator from the user and prints the appropriate result. This procedure is repeated till the user wishes to stop. The program should provide the user with three choices from Arithmetic operators, two choices from relational operators and two choices of increment and decrement operators and the ternary operator. **10**

c) Analyse the given program and write the expected output of the same. **05**

```
class Promote
{
      public static void main(String args[])
     {
         byte b = 42;
         char c = 'a';
         short s = 1024;
         int i = 50000;
         float f = 5.67f;
         double d = .1234;
         double result = (f * b) + (i / c) - (d * s);
         System.out.println((f * b) + " + " + (i / c) + " - " + (d * s));
          System.out.println("result = " + result);
        }
}
```

UNIT - II

2 a) Demonstrate overloading of methods in Java with an example program . **05**

b) Create a class SavingsAccount. Use a static class variable to store the annualInterestRate. Each object of the class contains a private instance variable savingsBalance indicating the balance amount in the account. Provide method calculateMonthlyInterest to calculate the monthly interest by multiplying the balance by annualInterestRate divided by 12; this interest should be added to savingsBalance. Provide a static method modifyInterestRate that sets the annualInterestRate to a new value. Write a driver program to test the class **08**

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.

SavingsAccount. Instantiate two different savingsAccount objects, saver1 and saver2, with balances of 2000.00 and 3000.00, respectively. Set annualInterestRate to 4%, then calculate the monthly interest and print the new balances for each of the objects .Then set the annualInterestRate to 5%, calculate the next month's interest and print the new balances for each of the objects. Print the expected output of the program.

c) Complete the program that uses command line arguments as inputs. The program adds the even numbers to sum1 and odd numbers to sum2 and displays the two resultants. Give appropriate comments for the significant statements and show the procedure to execute with command line arguments. **07**

```
class Commline
{
        public static void main(String args[])
        {
                int count, i=0,sum1=0,sum2=0;
                count= ------------------------------
                  …………
                 ………….
                ………….
        }
}
```

UNIT - III

3 a) Demonstrate any two uses of super keyword and two uses of final keyword with an appropriate program. **07**

b) Create a superclass, Student, and two subclasses, Undergrad and Grad. The superclass Student should have the following data members: name, ID, grade, age, and address. The superclass, Student should have at least one method: boolean isPassed (double grade). The purpose of the isPassed method is to take one parameter, grade (value between 0 and 100) and check whether the grade has passed the requirement for passing a course. In the Student class this method should be empty. The two subclasses, Grad and Undergrad, will inherit all data members of the Student class and override the method isPassed. For the UnderGrad class, if the grade is above 70.0, then isPassed returns true, otherwise it returns false. For the Grad class, if the grade is above 80.0, then isPassed returns true, otherwise returns false. Create a test class in which, create one Grad object and one Undergrad object. For each object, provide a grade and display the results of the isPassed method. 08

c) Rewrite the given program, fill in the missing statements to get the output as shown below. 05

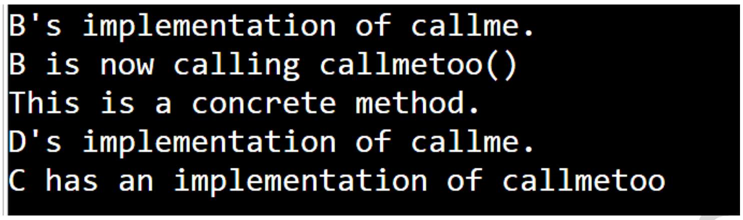
```
abstract class A
 {
          abstract void callme( );
          void callmetoo( ) { System.out.println("This is a concrete method."); }
}
class B extends A
{   /*Fill in the missing statements*/ }

abstract class C extends A
{/* Fill in the missing statements*/ }
```

```
class D extends C
{/* Fill in the missing statements*/ }
class AbstractDemo {
public static void main(String args[])
{
 // Fill in the missing statements
System.out.println( " B is now calling callmetoo()");
//Fill in the missing statements

}
```

EXPECTED OUTPUT:



```
B's implementation of callme.
B is now calling callmetoo()
This is a concrete method.
D's implementation of callme.
C has an implementation of callmetoo
```

UNIT - IV

4 a) Explain Generics with an example program. **05**

b) Write a program to design interface IntStack which contains two methods: push and pop( ). The interface should be implemented by two Stack classes i) FixedStack and ii) DynamicStack (Both classes contain integer array and top variable). FixedStack class will check for overflow condition and will not allow any more push operation in case of overflow. But DynamicStack, on occurrence of overflow condition will double the size of array inside stack class and continue with the push operation. **10**

c) Analyse the given erroneous program. Write the corrected program and underline the corrected statements. **05**

<u>sta.Java</u>

```
import Java.util.Scanner;
package staff;
class sta
{
    int sid;
    String snam;
    String dept;
    void getd( )
    {
        Scanner c=new Scanner(System.in);
        System.out.println("enter the staff id");
        sid=c.nextInt( );
        System.out.println("enter the name of the staff");
        snam=c.next( );
        System.out.println("enter the dept");
         dept=c.next( );
    }

    void putd( )
     { System.out.println("staff id="+sid+"staff name="+snam+"dept="+dept);  }
}
```

packdri.Java

```
class packdri
{
      public static void main(String args[])
      {
         sta ob2=new sta( );
         ob2.getd( );
         ob2.putd( );
      }
}
```

OR

5 a) Discuss about various access protection of variables in classes defined under packages in Java **05**

b) Create a program that demonstrates handling of userdefined exceptions. Create a class called "Employee" with the personal details like empid, empname, empage and empsalary. Two userdefined exceptions are raised when the following conditions are violated: **10**

i. The age of the employee should be between 21 and 58

ii. Minimum salary of an employee is 10000

c) Discuss about the alternate approach to implement multiple inheritance in Java. Justify the same with an example. **05**

UNIT - V

6 a) Differentiate between process based and thread based multitasking. **05**

b) Develop a Java program for creating multiple threads. First thread displays "Good Morning" every one second, the second thread displays "Hello" every two seconds and the third thread displays "Welcome" every three seconds. **10**

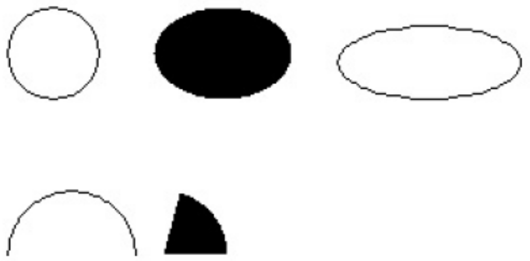
c) Discuss the two methods that let us convert between RGB and HSB. **05**

OR

7 a) Explain the delegation event model. **05**

b) Develop a program that gives the correct implementation of a producer and a consumer. **10**

c) Identify the graphics methods that help us draw the following shapes. Give the syntax of each of the methods and discuss. **05**



******