IC-1924

M.Sc. (INFORMATICS), II Sem – 2019

Paper IT-24 –Programming Languages

TIME: 03 hours Max Marks: 75

(Write your Roll No. on the top immediately on receipt of this question paper)

Q 1 is compulsory. Attempt any 4 questions from Q.2 to Q.6

Q1.

a. Provide output: (3)

```
public class Flipper {
        public static void main(String[] args) {
                String o = "-";
                switch("FRED".toLowerCase().substring(1,4)) {
                case "yellow":
                        o += "y";
                case "red":
                        o += "r";
                case "green":
                        o += "g";
                }
                System.out.println(o);
        }
}
```

b. Which is a correct class? (Choose all that apply with reason): (4)

Given:

```
public abstract interface Frobnicate { public void twiddle(String s); }

A. public abstract class Frob implements Frobnicate {
public abstract void twiddle(String s) { }
}
B. public abstract class Frob implements Frobnicate { }
C. public class Frob extends Frobnicate {
public void twiddle(Integer i) { }
}
D. public class Frob implements Frobnicate {
public void twiddle(Integer i) { }
}
E. public class Frob implements Frobnicate {
public void twiddle(String i) { }
public void twiddle(Integer s) { }
}
```

c. If the code compiles with following command line, what will be the result?

java Hilltop eyra vafi draumur kara (4)

```
public class HillTop {
        public static void main(String[] args) {
                String[] horses = new String[5];
                horses[4] = null;
                for(int i = 0; i < horses.length; i++) {
                        if(i < args.length)
                                horses[i] = args[i];
                        System.out.print(horses[i].toUpperCase() + " ");
                }
        }
}
```

d. Provide the output of following program? (4)

```
public class Tenor extends Singer {
        public static String sing() { return "fa"; }
        public static void main(String[] args) {
                Tenor t = new Tenor();
                Singer s = new Tenor();
                System.out.println(t.sing() + " " + s.sing());
        }
}
class Singer { public static String sing() { return "la"; } }
```

Q2.

a. Mention four pillars of object oriented programming? Which line will not compile? (5)

```
public class Fishing {
        byte b1 = 4;
        int i1 = 123456;
        long L1 = (long)i1; // line A
        short s2 = (short)i1; // line B
        byte b2 = (byte)i1; // line C
        int i2 = (int)123.456; // line D
        byte b3 = b1 + 7; // line E
}
```

b. Mention three restrictions that are applied to methods declared as static? Provide output of following: (5)

```
public class Mutant {
        public static void main(String[] args) {
                StringBuilder sb = new StringBuilder("abc");
                String s = "abc";
                sb.reverse().append("d");
                s.toUpperCase().concat("d");
                System.out.println("." + sb + ". ." + s + ".");
        }
}
```

c. What are variable-arity methods? Write the rules for creating variable-arity methods? Provide output of following code: (5)

```
import java.util.*;
public class Sequence {
        public static void main(String[] args) {
                ArrayList<String> myList = new ArrayList<String>();
                myList.add("apple");
                myList.add("carrot");
                myList.add("banana");
                myList.add(1, "plum");
                System.out.print(myList);
        }
}
```

Q3.

a. What is Dynamic Method Dispatch? Give example. What will the output of compiling and running following program? (5)

```
public class MyClass {
        public static void main(String[] args) {
                C c = new C();
                System.out.println(c.max(13, 29));
        }
}
class A {
        int max(int x, int y) { if (x>y) return x; else return y;
}
}
class B extends A{
        int max(int x, int y) { return super.max(y, x) - 10; }
}
class C extends B {
        int max(int x, int y) { return super.max(x+10, y+10); }
}
```

b. When a class hierarchy is created, in what order are the constructors for the classes that make up the hierarchy executed? Provide the output of following program? (5)

```
public class MyClass {
        public static void main(String[] args) {
                B b = new B("Test");
        }
}
class A {
        A() { this("1", "2"); }
        A(String s, String t) { this(s + t); }
        A(String s) { System.out.println(s); }
}
class B extends A {
        B(String s) { System.out.println(s); }
        B(String s, String t) { this(t + s + "3"); }
        B() { super("4"); };
}
```

c. Complete following table for Java access control: (3)

	Private	No Modifier	Protected	Public
Same class				
Same Package Subclass				
Same Package Non-Subclass				
Different package subclass				
Different package non-subclass				

d. What is static import? Which statements, when inserted at (1), will result in a program that prints 7, when compiled and run? (2)

```
// (1) INSERT ONE IMPORT STATEMENT HERE
public class RQ700_20 {
        public static void main(String[] args) {
                System.out.println(sqrt(49));}}
```

Q4.

a. How exception handling is done in Java? Provide the output of following program? (5)

```
import java.io.IOException;
import java.sql.SQLException;
class AllGoesWrong {
        public static void main(String[] args) {
                AllGoesWrong a = new AllGoesWrong();
                try {
                        a.blowUp();
                        System.out.println("a");
                } catch (IOException | SQLException e) {
                        System.out.println("c");
                } finally {
                        System.out.println("d");
                }
        }
        void blowUp() throws IOException, SQLException {
                throw new SQLException();
        }
}
```

b. Drawn and explain thread state transition diagram? Provide output of following: program? (5)

```
public class WaitTest {
        public static void main(String [] args) {
                System.out.print("1 ");
                synchronized(args){
                        System.out.print("2 ");
                        try {
                                args.wait();
                        }
                        catch(InterruptedException e){}
                }
                System.out.print("3 ");
        }
}
```

c. What are Deadlocks? What is need of synchronization in multithread programming? Write two ways to achieve synchronization in java? (5)

Q5.

a. Mention different type of I/O streams in java. Explain following line of code? (5)

```
BufferedReader br = new BufferedReader(new
InputStreamReader(System.in));
```

b. Explain Automatic resource management? Mention interface name required to support ARM? (5)

c. Write the use of volatile and transient modifiers? Write the use of finalize method? (5)

Q6.

a. What will the output of following program: (5)

```
public class MyClass {
        public static void main(String[] args) {
                B b = new C();
                A a = b;
                if (a instanceof A) System.out.println("A");
                if (a instanceof B) System.out.println("B");
                if (a instanceof C) System.out.println("C");
                if (a instanceof D) System.out.println("D");

        }
}
class A {}
class B extends A {}
class C extends B {}
class D extends C {}
```

b. Differentiate between: (5)

i. final and finally keywords.

ii. Method overloading and overriding.

c. Point out error in following program. Update the code to rectify the error. (5)

```
class Top {
        public Top(String s) { System.out.print("B"); }
}
public class Bottom2 extends Top {
        public Bottom2(String s) { System.out.print("D"); }
        public static void main(String [] args) {
                new Bottom2("C");
                System.out.println(" ");
        }
}
```