Variant 3

Part A

1. How does it called: representation of information about things?
2. How many ways are there to represent an algorithm?
3. How the instances of a class type are represented in memory: by value or by reference?
4. How many classes can we inherit from?
5. Can we overload static method?
6. Write the entities that can have 'final' modifier: instance field, static field, instance method, static method

Part B

1. What will be the output of this program?

```
public class ArraysInJava
{
    static void methodOne(int[] a)
    {
        int[] b = new int[5];

        a = b;

        System.out.print(a.length);

        System.out.print(b.length);
    }

    public static void main(String[] args)
    {
        int[] a = new int[10];

        methodOne(a);

        System.out.print(a.length);
    }
}
```

2. What will be the output of the following program?

```
class X
{
    static int x = 3131;

    static class Y
    {
        static int y = x++;

        static class Z
        {
            static int z = y++;
        }
    }
}

public class MainClass
{
    public static void main(String[] args)
    {
        System.out.println(X.x);

        System.out.println(X.Y.y);

        System.out.println(X.Y.Z.z);
    }
}
```

3. Is the below program written correctly? If yes, what will be the output?

```
abstract class A
{
    {
        System.out.println(1);
    }

    static
    {
        System.out.println(2);
    }
}

public class MainClass
{
    public static void main(String[] args)
    {
        A a = new A() { };
    }
}
```

4. What will be the output of the below program?

```
class A
{
    void methodA1(int i)
    {
        System.out.println(i+++i);
    }

    void methodA2(int i)
    {
        System.out.println(--i-i--);
    }
}

class B
{
    A a = new A()
    {
        void methodA1(int i)
        {
            System.out.println(++i+i++);
        }

        void methodA2(int i)
        {
            System.out.println(i---i);
        }
    };
}

public class MainClass
{
    public static void main(String[] args)
    {
        A a = new A();

        a.methodA1(10);

        a.methodA2(10);

        B b = new B();

        b.a.methodA1(10);

        b.a.methodA2(10);
```

```
    }
}
```

5. What will be the output of the following program?

```
class X
{
    {
        class Y
        {
            {
                System.out.println(1);
            }
        }

        new Y();
    }

    static
    {
        class Z
        {
            {
                System.out.println(2);
            }
        }

        new Z();
    }
}

public class MainClass
{
    public static void main(String[] args)
    {
        new X();
    }
}
```

6. What will be the output of the following program?

```
class A { }

class B extends A { }

class C extends B { }

interface ABC
{
    void method(A a);
}

interface PQR
{
    void method(B b);
}

class M implements ABC, PQR
{
    public void method(A a)
    {
        System.out.println(2);
    }

    public void method(B b)
    {
        System.out.println(3);
    }
}

public class MainClass
{
    public static void main(String[] args)
```

```
    {
        M m = new M();

        m.method(new A());

        m.method(new B());

        m.method(new C());
    }
}
```

7. What will be the output of the following program?

```
abstract class ABC
{
    abstract void methodOne();
}

abstract class XYZ extends ABC
{
    int i;

    @Override
    void methodOne()
    {
        methodOne(i *= i);
    }

    abstract void methodOne(int i);
}

class PQR extends XYZ
{
    public PQR(int i)
    {
        this.i = i;
    }

    @Override
    void methodOne(int i)
    {
        System.out.println(i++ * ++i);
    }
}

public class MainClass
{
    public static void main(String[] args)
    {
        PQR pqr = new PQR(1);

        pqr.methodOne();
    }
}
```

8. What will be the output of the below program?

```
class A
{
    void myMethod(Object o, Double D)
    {
        System.out.println(1);
    }

    void myMethod(Integer I, Number N)
    {
        System.out.println(2);
    }
}

class B extends A
{
    void myMethod(Float F, Double D)
```

```
    {
        System.out.println(3);
    }

    void myMethod(Double D, Integer I)
    {
        System.out.println(4);
    }
}

public class MainClass
{
    public static void main(String[] args)
    {
        B b = new B();

        b.myMethod(11.11, 0000);

        b.myMethod(8778, 3223);

        b.myMethod(2.3*1.2, 4.1*1.4);

        b.myMethod((float)23.56, 21.45);
    }
}
```