

## OOSD: Practice Problems 2

1. Complete this definition of a class each of whose objects encapsulate one natural number (nonnegative integer). Validate the input parameter for the constructor.

```
interface Ordered
{
    public boolean lessThan(Object other);
}
```

```
class Natural implements Ordered
{
    public Natural(int n)
    {
        :
    }

    public int getValue()           // return encapsulated number
    {
        :
    }

    public void squareNum()       // square encapsulated number
    {
        :
    }
}
```

2. Show what is printed when this Java program is executed.

```
public class Tester
{
    Tester(int n) { val = n; }
    int getVal() { return val; }
    void fix() { val = 2*val-1; }
    class Blue
    {
        void fix(int p) { val = p+val; }
    }
    Blue mkBlue() { return new Blue(); }
    class Orange
    {
        Orange(int k) { oran = k; }
        void fix() { val = oran*val; }
        void fix(int k) { oran = oran+k; }
        private int oran;
    }

    public static void main(String [] args)
    {
        Tester a = new Tester(1);
        Tester b = new Tester(2);
        Tester c = new Tester(3);

        Blue x = a.mkBlue();
        Orange y = b.new Orange(5);
        Orange z = c.new Orange(10);

        x.fix(7);    y.fix();    z.fix(2);

        System.out.println("a = " + a.getVal());
        System.out.println("b = " + b.getVal());
        System.out.println("c = " + c.getVal());

        a.fix();    b.fix();    c.fix();
        x.fix(3);    y.fix();    z.fix();

        System.out.println("a = " + a.getVal());
        System.out.println("b = " + b.getVal());
        System.out.println("c = " + c.getVal());
    }
    private int val;
}
```