

```
1 //Roman.java    MrG 2018.0314
2 public class Roman extends Object implements Comparable<Roman>
3 {
4     private int num;
5
6     public Roman(int num)
7     {
8         this.num = num;
9     }
10
11    public int getNum()
12    {
13        return num;
14    }
15
16    public int compareTo(Roman other)
17    {
18        return this.getNum()-other.getNum();
19    }
20
21    public boolean equals(Object other)
22    {
23        Roman temp = (Roman)other;
24        return temp.compareTo(this)==0;
25    }
26
27    public String toString()
28    {
29        String temp="";
30
31        int ones = num%10;
32        int tens = num/10;
33
34        if(tens==1){temp+="x";}
35        if(tens==2){temp+="xx";}
36        if(tens==3){temp+="xxx";}
37        if(tens==4){temp+="xl";}
38        if(tens==5){temp+="l";}
39        if(tens==6){temp+="lx";}
40        if(tens==7){temp+="lxx";}
41        if(tens==8){temp+="lxxx";}
42        if(tens==9){temp+="xc";}
43
44        temp+=" ";
45    }
```

```
46         if(ones==1){temp+="i";}  
47         if(ones==2){temp+="ii";}  
48         if(ones==3){temp+="iii";}  
49         if(ones==4){temp+="iv";}  
50         if(ones==5){temp+="v";}  
51         if(ones==6){temp+="vi";}  
52         if(ones==7){temp+="vii";}  
53         if(ones==8){temp+="viii";}  
54         if(ones==9){temp+="ix";}  
55  
56         return temp;  
57     }  
58  
59     public Roman add(Roman other)  
60     {  
61         return new Roman(num+other.getNum());  
62     }  
63  
64     public Roman sub(Roman other)  
65     {  
66         return new Roman(num-other.getNum());  
67     }  
68  
69     public Roman mul(Roman other)  
70     {  
71         return new Roman(num*other.getNum());  
72     }  
73 }
```