```java
//Roman2.java MrG 2018.0320
public class Roman2 extends Object implements Comparable<Roman2>
{
    private int[] digits;

    public Roman2(int num)
    {
        digits = new int[6];
        for(int i=0; i<6; i++)
        {
            digits[i]=num%10;
            num/=10;
        }

    }

    public int getNum()
    {
        int temp=0;

        for(int i=0; i<6; i++)
        {
            temp+=digits[i]*Math.pow(10,i);
        }

        return temp;
    }

    public int compareTo(Roman2 other)
    {
        return this.getNum()-other.getNum();
    }

    public boolean equals(Object other)
    {
        Roman2 temp = (Roman2)other;
        return temp.compareTo(this)==0;
    }

    public String toString()
    {
        String temp="";

        if(digits[5]==1){temp+="C";}
        if(digits[5]==2){temp+="CC";}
```

```
46          if(digits[5]==3){temp+="CCC";}
47          if(digits[5]==4){temp+="CD";}
48          if(digits[5]==5){temp+="D";}
49          if(digits[5]==6){temp+="DC";}
50          if(digits[5]==7){temp+="DCC";}
51          if(digits[5]==8){temp+="DCCC";}
52          if(digits[5]==9){temp+="CM";}
53
54          temp+=" ";
55
56          if(digits[4]==1){temp+="X";}
57          if(digits[4]==2){temp+="XX";}
58          if(digits[4]==3){temp+="XXX";}
59          if(digits[4]==4){temp+="XL";}
60          if(digits[4]==5){temp+="L";}
61          if(digits[4]==6){temp+="LX";}
62          if(digits[4]==7){temp+="LXX";}
63          if(digits[4]==8){temp+="LXXX";}
64          if(digits[4]==9){temp+="XC";}
65
66          temp+=" ";
67
68          if(digits[3]==1){temp+="m";}
69          if(digits[3]==2){temp+="mm";}
70          if(digits[3]==3){temp+="mmm";}
71          if(digits[3]==4){temp+="mV";}
72          if(digits[3]==5){temp+="V";}
73          if(digits[3]==6){temp+="Vm";}
74          if(digits[3]==7){temp+="Vmm";}
75          if(digits[3]==8){temp+="Vmmm";}
76          if(digits[3]==9){temp+="mX";}
77
78          temp+=" ";
79
80          if(digits[2]==1){temp+="c";}
81          if(digits[2]==2){temp+="cc";}
82          if(digits[2]==3){temp+="ccc";}
83          if(digits[2]==4){temp+="cd";}
84          if(digits[2]==5){temp+="d";}
85          if(digits[2]==6){temp+="dc";}
86          if(digits[2]==7){temp+="dcc";}
87          if(digits[2]==8){temp+="dccc";}
88          if(digits[2]==9){temp+="cm";}
89
90          temp+=" ";
```

```
91
92          if(digits[1]==1){temp+="x";}
93          if(digits[1]==2){temp+="xx";}
94          if(digits[1]==3){temp+="xxx";}
95          if(digits[1]==4){temp+="xl";}
96          if(digits[1]==5){temp+="l";}
97          if(digits[1]==6){temp+="lx";}
98          if(digits[1]==7){temp+="lxx";}
99          if(digits[1]==8){temp+="lxxx";}
100         if(digits[1]==9){temp+="xc";}
101
102         temp+=" ";
103
104         if(digits[0]==1){temp+="i";}
105         if(digits[0]==2){temp+="ii";}
106         if(digits[0]==3){temp+="iii";}
107         if(digits[0]==4){temp+="iv";}
108         if(digits[0]==5){temp+="v";}
109         if(digits[0]==6){temp+="vi";}
110         if(digits[0]==7){temp+="vii";}
111         if(digits[0]==8){temp+="viii";}
112         if(digits[0]==9){temp+="ix";}
113
114         return temp;
115     }
116
117     public Roman2 add(Roman2 other)
118     {
119         return new Roman2(getNum()+other.getNum());
120     }
121
122     public Roman2 sub(Roman2 other)
123     {
124         return new Roman2(getNum()-other.getNum());
125     }
126
127     public Roman2 mul(Roman2 other)
128     {
129         return new Roman2(getNum()*other.getNum());
130     }
131 }
```