```
public abstract class Circuit{
    char method;
    double resistance;
    Circuit resistor0;
    Circuit resistor1;
    public abstract double resistance();
    public Circuit(){

    }
    public static Circuit getCircuit(String net){
        if(net.length()==1) return new Single(net);
        if(net.charAt(1)=='-'){
            return new Parallel(net);
        }
        else if(net.charAt(1)=='/'){
            return new Series(net);
        }
        else{
            return new Single(net);
        }
    }
    public static int getMidLength(String net){
        char[] subNet = net.substring(3, net.length()-1).toCharArray();// (/,(-,(-,2,4),6),7) , (/,3.4,2.7)
        int stack = 0;
        int counter = 0;
        if(subNet[0]!='('){
            while(subNet[counter]!=',') counter++;
            return counter;
        }
        for (counter = 0; counter < subNet.length; counter++) {
            if(subNet[counter]=='(') stack++;
            if(subNet[counter]==')') stack--;
            if(stack == 0) break;
        }
        return counter+1;
    }
    @Override
    public String toString() {
        if(this.method=='s') return String.valueOf(this.resistance);
        return String.format("(%c,%s,%s)", this.method,this.resistor0.toString(),this.resistor1.toString());
    }
}

class Parallel extends Circuit{
    public Parallel(String net) {
        this.method = '-';
        int midLength = getMidLength(net);
        int endLength = net.length()-midLength-5;
        this.resistor0 = Circuit.getCircuit(net.substring(3, 3+midLength));
        this.resistor1 = Circuit.getCircuit(net.substring(4+midLength,net.length()-1));
        this.resistance = 1/((1/resistor0.resistance)+(1/resistor1.resistance));
    }
    public double resistance(){
        return this.resistance;
    }
}

class Series extends Circuit{
    public Series(String net){
        this.method = '/';
        int midLength = getMidLength(net);
        this.resistor0 = Circuit.getCircuit(net.substring(3, 3+midLength));
        this.resistor1 =
Circuit.getCircuit(net.substring(4+midLength,net.length()-1));
        this.resistance = this.resistor0.resistance+this.resistor1.resistance;
    }
    public double resistance(){
        return this.resistance;
    }

}

class Single extends Circuit{
    public Single(String net){
        this.method = 's';
        this.resistance = Double.parseDouble(net);
    }
    public double resistance(){
        return this.resistance;
    }
}
class TestCircuit{
    public static void main(String[] args) {
        String net = "(-,(/,(-,(-,2,4),6),7),(/,3.4,2.7))";
        Circuit circuit = Circuit.getCircuit(net);
        System.out.println(circuit.toString());
        System.out.println(circuit.resistance());
    }
}
```