状态模式（State），当一个对象的内在状态改变时允许改变其行为，这个对象看起来像是改变了其类。

优点：

1. 封装了转换规则；
2. 枚举可能的状态，在枚举状态之前需要确定状态种类；
3. 将所有与某个状态有关的行为放到一个类中，并且可以方便地增加新的状态，只需要改变对象状态即可改变对象的行为；
4. 允许状态转换逻辑与状态对象合成一体，而不是某一个巨大的条件语句块；
5. 可以让多个环境对象共享一个状态对象，从而减少系统中对象的个数。

缺点：

1. 状态模式的使用必然会增加系统类和对象的个数；
2. 状态模式的结构与实现都较为复杂，如果使用不当将导致程序结构和代码的混乱；
3. 状态模式对"开闭原则"的支持并不太好，对于可以切换状态的状态模式，增加新的状态类需要修改那些负责状态转换的源代码，否则无法切换到新增状态，而且修改某个状态类的行为也需修改对应类的源代码。

运行结果：

