

```
1  enum ballPhase {
2      HOME, betweenTrap, waitingLowerTrap, beforePusherPhase, readingPusherPhase,
3      whiteBallFall, blackBallFall, inTheBox
4  }
5
6  public class Ball extends Actor implements
Runnable {
7      private ballPhase phase;
8      private int x0, y0;
9      protected Terminal<Boolean> upperTrap=new Terminal<Boolean>(false);
10     protected Terminal<Boolean> lowerTrap=new Terminal<Boolean>(false);
11     protected Terminal<Boolean> pusher=new Terminal<Boolean>(false);
12     protected Terminal<Integer> ballColor=new Terminal<Integer>(0);
13     protected Terminal<Boolean> whiteBall=new Terminal<Boolean>(false);
14     protected Terminal<Boolean> blackBall=new Terminal<Boolean>(false);
15     private int powerOnDelay=200;
16
17     public Ball(JSONObject src) {
18         super(src);
19         x0=x;
20         y0=y;
21         home();
22     }
23
24
```

```
25 void home() {
26     x=x0;
27     y=y0;
28     index=(int) random(100)%2;
29     phase=ballPhase.HOME;
30     ballColor.write(index);
31     blackBall.write((index==0));
32     whiteBall.write(!(index==0));
33 }
34
35 void run() {
36     for (;;) {
37         if (powerOnDelay>0) {
38             powerOnDelay--;
39             visible.write(powerOnDelay==0);
40         } else
41             switch(phase) {
42                 case HOME:
43                     if (upperTrap.read())
44                         phase=ballPhase.betweenTrap;
45                     break;
46                 case betweenTrap:
47                     if (y<y0+64)
48                         y++;
```

```
49         else
50             phase=ballPhase.waitingLowerTrap;
51         break;
52     case waitingLowerTrap:
53         if (lowerTrap.read())
54             phase=ballPhase.beforePusherPhase;
55         break;
56     case beforePusherPhase:
57         if (y<y0+176)
58             y++;
59         else
60             phase=ballPhase.readingPusherPhase;
61         break;
62     case readingPusherPhase:
63         if (pusher.read())
64             phase=ballPhase.whiteBallFall;
65         else {
66             y++;
67             if (y>325)
68                 phase=ballPhase.blackBallFall;
69         }
70         break;
71     case whiteBallFall:
72         x--;
```

```
73         y++;
74         if (y>520)
75             phase=ballPhase.inTheBox;
76         break;
77     case inTheBox:
78         home();
79         break;
80     case blackBallFall:
81         x++;
82         y++;
83         if (y>520)
84             phase=ballPhase.inTheBox;
85         break;
86     }
87     try {
88         java.lang.Thread.sleep(samplingTime);
89     }
90     catch (Exception e) {
91         println(e);
92     }
93 }
94 }
95
96
```

C:/nqCbtp/JavaApplication8/src/Terminal.java

```
97     public Terminal getTerminal(String terminalName) {
98         if (terminalName.equals("upperTrap"))
99             return upperTrap;
100     else
101         if (terminalName.equals("lowerTrap"))
102             return lowerTrap;
103     else
104         if (terminalName.equals("pusher"))
105             return pusher;
106     else
107         if (terminalName.equals("ballColor"))
108             return ballColor;
109     else
110         if (terminalName.equals("whiteBall"))
111             return whiteBall;
112     else
113         if (terminalName.equals("blackBall"))
114             return blackBall;
115     return super.getTerminal(terminalName);
116 }
117 }
```