```
// Eugene Ngo
     // 12/2/2022
 3
     // EE 271
 4
     // Lab 5
 6
     // centerLight module takes in clk, reset, L, R, NL, NR, restart as inputs and they are all
     1-Bit and returns
 7
     // lightOn which is also 1-Bit. This module determines the next state based on the inputs
     and moves the light
 8
     // to the left if key 3 is pressed and moves the light to the right if key 0 is pressed.
     When the game is reset
 9
     // or restarted then this module recenters the light for the next game.
10
11
     module centerLight (lightOn, clk, reset, L, R, NL, NR, restart);
12
         output logic lightOn;
13
         input logic clk, reset;
14
         input logic L, R, NL, NR, restart;
15
16
17
        enum {on, off} ps, ns;
18
        always_comb begin
19
            case(ps)
20
                       if(\sim R \& L \mid R \& \sim L) \text{ ns = off};
               on:
21
                       else ns = on;
22
23
24
               off:
                       if(NR \& L \& \sim R \mid NL \& R \& \sim L) ns = on;
                       else ns = off;
25
26
            endcase
27
28
29
        end
        always_comb begin
30
31
32
33
            case(ps)
               on: lightOn = 1;
               off: lightOn = 0;
34
            endcase
35
        end
36
37
         always_ff @(posedge clk) begin
            if(reset | restart)
38
39
               ps \ll on;
40
            else
41
               ps <= ns;
42
        end
43
44
     endmodule
45
46
47
     // centerLight_testbench tests all expected, unexpected, and edgecase behaviors
     module centerLight_testbench();
48
         logic L, R, NL, NR;
49
         logic lightOn;
50
         logic clk, reset;
51
52
53
54
        centerLight dut (.lightOn, .clk, .reset, .L, .R, .NL, .NR);
        //Set up the clock
55
        parameter CLOCK_PERIOD = 100;
56
57
         initial begin
            clk <= 0;
58
            forever #(CLOCK_PERIOD / 2)
59
            clk \ll \sim clk;
60
        end
61
         //Set up the inputs to the design. Each line is a clock cycle
62
63
         initial begin
64
                                                    @(posedge clk)
                                                    @(posedge clk)
65
            reset \leftarrow 1;
                                                    @(posedge clk)
66
67
                                                    @(posedge clk)
            reset \leftarrow 0;
68
                                                    @(posedge clk)
69
            L <= 1; R <= 0; NL <= 0; NR <= 1;
                                                    @(posedge clk)
70
                                                    @(posedge clk)
```

```
@(posedge clk)
 71
72
73
74
75
76
77
                                                                 @(posedge clk)
                                     NL \ll 1; NR \ll 0;
                                                                 @(posedge clk)
                                                                @(posedge clk)
@(posedge clk)
@(posedge clk)
@(posedge clk)
@(posedge clk)
                L <= 0; R <= 1;
 78
79
80
                                                                 @(posedge clk)
                                                                 @(posedge clk)
 81
82
83
84
                                     NL \leftarrow 0; NR \leftarrow 1;
                                                                 @(posedge clk)
                                                                 @(posedge clk)
                                                                 @(posedge c]k)
                                                                 @(posedge clk)
                                                                 @(posedge clk)
 85
                L \ll 1;
                                     NL \ll 1; NR \ll 0;
 86
87
                                                                 @(posedge clk)
                                                                 @(posedge clk)
 88
89
90
                                                                 @(posedge clk)
                                                                @(posedge clk)
@(posedge clk)
@(posedge clk)
                reset \leftarrow 1;
 91
 92
                                                                 @(posedge clk)
                L \le 0; R \le 1; NL \le 1; NR \le 0;
 93
                                                                 @(posedge clk)
 94
                                                                 @(posedge clk)
 95
                                                                 @(posedge clk)
                L \le 1; R \le 0; NL \le 0; NR \le 0;
 96
                                                                 @(posedge clk)
 97
                                                                 @(posedge clk)
                                                                 @(posedge c]k)
 98
                L \le 0; R \le 1; NL \le 0; NR \le 0;
 99
                                                                 @(posedge clk)
100
                                                                 @(posedge clk)
101
                $stop;
102
            end
103
        endmodule
```