```
1   // Eugene Ngo
2   // 1/20/2023
3   // EE 371
4   // Lab 2 Task 3
5
6   // the seg 7 module is for a single hex display. It takes in a 4-bit value and display the
    corresponding
7   // hexadecimal value on the hex display.
8
9   module seg7 (in, out);
10
11     input logic [3:0] in;
12     output logic [6:0] out;
13
14     // Initiating a hex ram to drive the hex displays
15     logic [15:0][6:0] hex_ram;
16     assign hex_ram[0]  = 7'b1000000; // 0
17     assign hex_ram[1]  = 7'b1111001; // 1
18     assign hex_ram[2]  = 7'b0100100; // 2
19     assign hex_ram[3]  = 7'b0110000; // 3
20     assign hex_ram[4]  = 7'b0011001; // 4
21     assign hex_ram[5]  = 7'b0010010; // 5
22     assign hex_ram[6]  = 7'b0000010; // 6
23     assign hex_ram[7]  = 7'b1111000; // 7
24     assign hex_ram[8]  = 7'b0000000; // 8
25     assign hex_ram[9]  = 7'b0010000; // 9
26     assign hex_ram[10] = 7'b0001000; // a
27     assign hex_ram[11] = 7'b0000011; // b
28     assign hex_ram[12] = 7'b1000110; // c
29     assign hex_ram[13] = 7'b0100001; // d
30     assign hex_ram[14] = 7'b0000110; // e
31     assign hex_ram[15] = 7'b0001110; // f
32
33     assign out = hex_ram[in];
34
35  endmodule
36
37  // seg7_testbench tests all expected, unexpected, and edgecase behaviors
38  module seg7_testbench();
39     logic [3:0] in;
40     logic [6:0] out;
41
42     seg7 dut (.in, .out);
43
44     initial begin
45     // Test some random inputs
46        in = 4'h0; #10;
47        in = 4'h3; #10;
48        in = 4'h8; #10;
49        in = 4'hb; #10;
50        in = 4'hf; #10;
51        $stop;
52     end
53  endmodule
```