```
1   // Eugene Ngo
2   // 12/2/2022
3   // EE 271
4   // Lab 5
5
6   // comparator takes 1-Bit clk, 1-Bit reset, 10-Bit A, and 10-Bit B as inputs and returns
    1-Bit value_final
7   // This module determines the difficulty and value of the CPU input by comparing the switch
    value with
8   // the LFSR random value
9
10  module comparator(clk, reset, A, B, value_final);
11     output logic value_final;
12     input logic clk, reset;
13     input logic [9:0] A, B;
14
15     logic value;
16
17     // consider metastability for computer
18        always_comb begin
19
20            value = (A > B); // where: A = SW , B = LFSR
21
22        end
23
24     always_ff @(posedge clk) begin
25        value_final <= value;
26     end
27
28  endmodule
29
30  // comparator_testbench tests all expected, unexpected, and edgecase behaviors
31  module comparator_testbench();
32     logic value_final;
33     logic clk, reset;
34     logic [9:0] A, B;
35
36     comparator dut(.clk, .reset, .A, .B, .value_final);
37
38     parameter CLOCK_PERIOD = 100;
39     initial begin
40        clk <= 0;
41        forever #(CLOCK_PERIOD / 2)
42        clk <= ~clk;
43     end
44
45     initial begin
46           A = 10'b0010000001; B = 10'b0100000010;   @(posedge clk);
47                                                     @(posedge clk);
48                               B = 10'b0000010000;   @(posedge clk);
49                                                     @(posedge clk);
50        $stop;
51     end
52  endmodule
53
54
```