```systemverilog
// Alan Li
// 01/15/2022
// EE 371
// Lab #1

// hexDisplay takes 5-bit inputCount as inputs and return 7-bits HEX5, HEX4, HEX3, HEX2, HEX1, HEX0 as output.
// Upon start, the HEX display will display "clear0", when enter signal is given, the HEX display the counter on HEX1 and HEX0. The counter will keep adding up until reach 25.
// Then the hex display will diaplay "FULL25".
// If the counter reach 0, the hex display will display "clear0" as there is no car in the parking lot

module hexDisplay(clk, inputCount, HEX5, HEX4, HEX3, HEX2,HEX1, HEX0);
   input clk;
   input logic [4:0] inputCount;
   output logic [6:0] HEX5, HEX4, HEX3, HEX2, HEX1, HEX0;

   // 7-bit parameter for the display to display different numbers or characters
   //                            6543210
   parameter [6:0]   zero =   7'b1000000,
                     one =    7'b1111001,
                     two =    7'b0100100,
                     three =  7'b0110000,
                     four =   7'b0011001,
                     five =   7'b0010010,
                     six =    7'b0000010,
                     seven =  7'b1111000,
                     eight =  7'b0000000,
                     nine =   7'b0011000,
                     F =      7'b0001110, //FULL
                     U =      7'b1000001,
                     L =      7'b1000111,

                     C =      7'b1000110, //CLEAR
                     E =      7'b0000110,
                     A =      7'b0001000,
                     R =      7'b1001100,
                     blk =    7'b1111111;


   // The hex are driven off upon start.
   // The hex will display counters when enter or exit signal is given
   // When the counter is 0, it will display "clear0". When full it will display "full25"
   always_comb begin
      HEX5 = blk;
      HEX4 = blk;
      HEX3 = blk;
      HEX2 = blk;
      HEX1 = blk;
      HEX0 = blk;
      case(inputCount)
         0: begin HEX5 = C; HEX4 = L; HEX3 = E; HEX2 = A; HEX1 = R; HEX0 = zero; end
         1: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk; HEX1 =
blk; HEX0 = one; end
         2: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk; HEX1 = blk; HEX0 = two; end
         3: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk; HEX1 = blk; HEX0 = three; end
         4: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk; HEX1 = blk;HEX0 = four; end
         5: begin HEX5 = F; HEX4 = U; HEX3 = L; HEX2 = L;HEX1 = blk; HEX0 = five; end // for demo purpose

         //5: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk; HEX1 = blk;HEX0 = five; end
         6: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk; HEX1 = blk;HEX0 = six; end
         7: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk; HEX1 = blk;HEX0 = seven; end
         8: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk; HEX1 = blk;HEX0 = eight; end
         9: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk; HEX1 = blk;HEX0 = nine; end
         10: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = one; HEX0 = zero; end
         11: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = one; HEX0 = one; end
         12: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = one; HEX0 = two; end
         13: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = one; HEX0 = three; end
         14: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = one; HEX0 = four; end
         15: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = one; HEX0 = five; end
         16: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = one; HEX0 = six; end
         17: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = one; HEX0 = seven; end
         18: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = one; HEX0 = eight; end
         19: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = one; HEX0 = nine; end
         20: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = two; HEX0 = zero; end
         21: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = two; HEX0 = one; end
         22: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = two; HEX0 = two; end
         23: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = two; HEX0 = three; end
         24: begin HEX5 = blk; HEX4 = blk; HEX3 = blk; HEX2 = blk;HEX1 = two; HEX0 = four; end
         25: begin HEX5 = F; HEX4 = U; HEX3 = L; HEX2 = L;HEX1 = two; HEX0 = five; end
      endcase
   end

endmodule

// counter_testbench tests all expected behavior that the parking lot
```

```
occupancy counter system in the lab may encounter
module hexDisplay_testbench();
   logic clk;
   logic [4:0] inputCount;
   logic [6:0] HEX5, HEX4, HEX3, HEX2, HEX1, HEX0;

   hexDisplay dut (.clk(clk), .inputCount(inputCount), .HEX5(HEX5), .HEX4(HEX4), .HEX3(HEX3), .HEX2(HEX2), .HEX1(HEX1), .HEX0(HEX0));

   parameter CLOCK_PERIOD = 100;

   initial begin
      clk <= 0;
      forever #(CLOCK_PERIOD/2) clk <= ~clk;
   end

   initial begin
      inputCount <= 0;  @(posedge clk);
      inputCount <= 10; @(posedge clk);
      inputCount <= 20; @(posedge clk);
      inputCount <= 25; @(posedge clk);
      $stop;
   end
endmodule
```