Exercise FPGA-1: Comparison Operator (5 Points)

Evaluate the following comparisons and mark them with “true” and “false”.

```
4'b1010    ==    4'b1011

4'b11xx    >     4'b10xx

4'b1100    <     4'b1100

4'b11xx    ===   4'b10zz

4'b110z    ===   4'b110z
```

Exercise FPGA-2: Shift Register (10 Points)

Write the Verilog source code for the implementation of a simple 1-bit shift register. The memory depth should be 4 Bit. The shift register should shift from MSB (MostSignificantBit) to LSB (LeastSignificantBit) (from left to right). Only the last Bit should be connected to the output.

The functionality of the shift register should be activated and deactivated by the Signal EN.

Action should take place at the rising edge of the clock.

```
module gates( input CLK, input EN, input IN, output OUT)

endmodule
```

Exercise FPGA-3: BCD encoder (10 Points)

Write the Verilog source code for the implementation of a BCD encoder. The input is a 7 bit binary number (decimal range 0 – 99). The output should be 2 digits (each 4 bit wide) of the binary coded decimals (ones and tens).

```
module gates(input[6:0] decimal, output[3:0] tens, output[3:0] ones)
    integer i;




endmodule
```

Excercise FPGA-4: Evaluation of FPGA counter (10 Points)

Given is the following Verilog-Code:

```
module counter(   input CLK,
                  output reg [1:0] cnt2,
                  output reg [1:0] cnt1 )

  initial begin
    cnt1 = 0;
    cnt2 = 0;
  end

  always @(negedge CLK) begin
    cnt1 <= cnt1 + 'b1;
    if(cnt1 > 'b10) begin
      cnt1 = 'b10;
    end
  end

  always @(posedge CLK) begin
    cnt2 = cnt2 + 'b1;
    if(cnt2 > 'b10) begin
      cnt2 = 'b10;
    end
  end

endmodule
```

Evaluate for five sequential clock cycles the value of the counters `cnt1` and `cnt2` .
Each clock cycle consists of

`0:` low (logic 0)
`r:` rising edge (transisition 0 to 1)
`1:` high (logic 1)
`f:` falling edge (transisition 1 to 0)