Midterm review

Or,
Everything I need to know I learned in ECE 4514.
(well, almost...)

General format

- Roughly 30% will be short answer
- The rest will be long answer
- Closed book, closed notes, closed neighbor
Topics

- Design process
- Entities, architectures, processes
  - Structural, dataflow, algorithmic descriptions
  - Sensitivity lists, wait statements, sequential statements
  - Signal, variable assignments
- Concurrent/sequential signal assignments
- Identifiers
- Data types: Literals, built-in types, enumerated types, arrays, records, etc.
- Assert statements
- Attributes
- Subprograms: Procedures, functions
- Waveform drivers, types of delay, waveform updating
- Macro/micro time
- Packages/libraries

Things you should be able to do

- Examples:
  - Given two architectures, do they give the same behavior?
  - Write a function/procedure/architecture that does X. (Roughly 10-30 lines of code)
    - Syntax matters.
  - Draw a schematic given a structural model.
  - Show a waveform or sequence of events given a piece of code and a set of inputs.
  - Given an incomplete piece of code, fill in the missing lines.
Examples: short

- An _______________________ description defines the I/O response but does not imply any particular physical implementation, while in a _______________________ description the data dependencies in the description match those in a real implementation.

- What are the two types of association used with port maps? Give an example of each one.

Examples: short

- Tell if the two architectures are equivalent, i.e. do they give exactly the same behavior for the entity? Briefly explain why or why not.

architecture first of delay is
begin
    process
        variable a1, a2, a3; std_logic:= '0';
        begin
            a1 := not B;
            a2 := B and C;
            a3 := a1 and D;
            A <= a1 or a2 or a3;
            wait on B, C, C;
        end process;
    end first;

architecture second of delay is
begin
    process (B, C, D)
        variable a1, a2, a3; std_logic:= '0';
        begin
            a1 := not B;
            a2 := B and C;
            a3 := a1 and D;
            A <= a1 or a2 or a3;
            wait on B, C, D;
        end process;
    end second;
Examples: Long

Design a device that counts the number of ones in a byte. The device receives as input an 8 bit word called \texttt{code8}, and outputs a 4-bit vector called \texttt{count}. The leftmost bit of \texttt{count} is the most significant bit, e.g., if there 8 ones in \texttt{code8} then \texttt{count} = “1000”. The entity declaration is given for you. Give an architecture implementing the entity. Your code should be efficient and easily understandable. You may use any of the VHDL constructs that we discussed in class.

```vhdl
library ieee;
use ieee.std_logic_1164.all;
entity bean_counter is
    port (code8: in std_logic_vector(7 downto 0);
          count: out std_logic_vector(3 downto 0));
end bean_counter;
```