COSC-163: Introduction to Computer Architecture
Fall 2019
Sample exam

1. Show the design of a *4-to-1 multiplexer*. That is, express the logic and draw a circuit for a device that would take four 1-bit data inputs (A_0, A_1, A_2, A_3), as well as a 2-bit selection $S = (S_1, S_0)$, and emits an output Y that is the A_S input. Show this circuit using only basic gates.

2. When we devised how a basic memory element would work, we came up with the following truth table:

Q	R	S	Q
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	x
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	x

   We later used the *cross-coupled* NOR *gates* to implement this function and devise our S-R latch. **Using Boolean algebra, prove that the two are equivalent. Note:** If you have difficulty devising an algebraic proof, you may prove the equivalence using any other legitimate method, but for a partial loss of credit.

3. Consider our multiplier circuit from Lab 5. It handles *unsigned integers* correctly, but what if we want it **also to handle signed, two's-complement integers**? It doesn't handle them properly as it's currently designed, so:

   Describe and sketch how you would change or augment our multiplier to handle two signed integers as inputs, and then produce the correct signed output. If you draw the multiplier, you need not draw its every detail, but only those that are altered or affected by your changes.

4. Answer the following questions about *pipelined processors*.
   (a) How does a *critical path* limit the clock rate of a processor?
   (b) How does a pipelined datapath addresses this critical-path problem, allowing for higher clock rates?
   (c) What is a *data hazard* and how does a pipelined processor handle them?
   (d) What is a *control hazard* and how does a pipelined processor handle them?