1) When is the first exam? Is it close-book, and closed notes? Will there be four different versions, so that the person next to you has different questions? Which one will be get you past a job interview – knowing what you’re doing, or having a high GPA?

2) Convert the following integers into 8-bit, unsigned numbers: 5, 27, 43. Now show the 8-bit 2’s complement representations for -5, -27, -43. In 2’s complement binary, add 10110011 and 00101011. What is the result of the addition in decimal?

3) Convert the hex number xA2 into decimal, assuming that it represents an 8-bit unsigned number. What is it in decimal if it’s an 8-bit 2’s complement number?

4) Add hex numbers x16 and x99, and show the result in hex.

5) With 5-bit 2’s complement numbers.... What is 01101 – 00111? **SHOW YOUR WORK**, and only use addition, bit flipping, and adding of 1 to get your result. You know how you can make a number negative in 2’s complement, right? This is a question checking if you can do that.

6) Sketch a 2-input, 4-output decoder using Boolean logic gates.

7) Using two OR gates, use a 2-input, 4-output decoder to implement two different Boolean logic functions.
   Function 1: (A\!\!\!\!\!\!\!\!\!\!B) + (\!A \!\!\!\!\!\!\!\!\!\!B)
   Function 2: (\!A B) + (A \!\!\!\!\!\!\!\!\!\!B) + (AB)

8) Sketch the following: a 2-input, 1-output Mux (use Boolean logic gates)

9) Show the truth table for a full adder circuit (it takes inputs A, B, and Cin, and produces outputs Cout and S).

10) Assume you have a four boxes, and each has a full adder circuit inside. Arrange and connect them so that you can add two four-bit numbers together.

11) Draw an RS latch. Draw a gated D-Latch. (use Boolean logic gates)

12) Draw an inverter (not gate) at the transistor level. Make sure your sketch shows which transistor is P, and which is N.

13) Build a truth table for the circuit shown.

```
<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>AND</th>
<th>OR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

```
14) Build a truth table for the following Finite State Machine. Show the input state bits S1 and S0, where you go in the next state S1’ and S0’, and the lights that you turn on or off, A and B.

15) What is the sequence of lights if you start in state 00? Just show what happens for 6 steps (the machine will run forever, blinking lights on and off).

16) In the LC3, how many registers are there?

17) What does IR stand for? PC? ALU?

18) The LC3 ALU supports ADD, AND, and NOT mathematical operations (and there are circuits for these in the ALU). What is the name of circuit that selects the different outputs, so that you can save the value you want back into a register file? And how many bits would you need to have this circuit perform the selection?

19) If the LC3 ALU performs a “NOT” operation on the binary number 10110111, what is the output?

20) If the LC3 ALU performs an “AND” operation on the binary numbers 11001101 and 10101011, what is the output?

21) Build a truth table for the binary function NOT( (NOT A) AND (NOT B)). In a program, using ! for “NOT”, this might look like !(A & !B). The & is frequently used to mean “AND,” applied on a bit-by-bit basis to a binary number.

22) What’s the name of the movie that has Benedict Cumberbatch playing Alan Turing?