1. Let's see how delays in a circuit can affect the circuit's output. In VHDL, you can assign delay to a signal in many ways. One is with a statement like the following:

\[ F \leftarrow A \text{ OR } B \text{ after } 1 \text{ ns}; \]

This assigns \( F \) the value of the output of the OR gate with a 1 ns delay from when the last input is stable. If we assign each gate a 1 ns delay, we would call this a unit delay simulation. This is useful to start to see how delays can cause us problems and unexpected outputs. In reality, AND gates and OR gates might have different delays and also different delays from certain inputs to the output. We ignore all these issues but the resulting simulation is still interesting.

With that said, here's an interesting problem: Design a 3-bit multiplier that will multiply two, 3-bit unsigned, binary inputs and produce a 6-bit, unsigned binary output. For your multiplier, you should use a couple of 3-bit full-adders that you need to design. In designing these adders, use statements like above to add a unit delay to the output of the \( S \) (sum) and \( C_{out} \) (carry-out) bits. That is, all of the outputs should change 1 ns after the inputs change.

After wiring up your multiplier, simulate it with inputs that find the square of all possible 3-bit numbers (that is, \( 0^2 \), \( 1^2 \), \( 2^2 \) and so on). After changing the inputs, run the simulation for 10 ns before changing the inputs, again. Comment on what happens at the outputs and why. Prove that your circuit acts as you expect by discussing the longest path through your circuit when the inputs change from 5 to 6.

Also, you might find it helpful to change the default radix to unsigned so that you can look at things, and type them, as decimal numbers (not a series of bits). You can do this in the Simulate→Simulate Options menu. Now, when you enter the value of your inputs, you can use 0, 1, 2, etc. as opposed to 000, 001, 010, etc.

Turn in your source code for both the adder and the multiplier. Also turn in your simulation waveforms with a zommed-in version for the inputs changing from 5→6 that clearly shows the delays and what the outputs do at this point.

1. - 1 -

2.
2. Another interesting display of what delay can do is to look at the d-latch discussed in class. Write a VHDL structural model of the d-latch using a NAND2 and INV gate with 1 ns, each (again, unit delay). Wire these up and see what happens when you change the D input 1 ns before the CLK input goes high. Convince yourself this makes sense...

(Note, since this circuit has feedback in it you will have a problem if you define the Q and Qbar signals as outputs. The reason is that an output cannot feedback to be an input to anything else in this component. An easy way around this is to define the Q and Qbar signals as inout which is a bi-directional signal and will allow the feedback you need.)