CSE4372/5372 (Spring 2025)
Lab #3

This lab is due by February 20, with a 20% penalty per week day for being late.

In this lab, you will write the register file module. This module will be accessed by the ID and WB stages of the pipeline.

The following steps will guide you through this process:

1. Create a module, rv32i_regs, with the following port list:

```
// system clock and synchronous reset
input clk,
input reset,

// inputs
input [4:0]     rs1_reg,
input [4:0]     rs2_reg,
input           wb_enable,
input [4:0]     wb_reg,
input [31:0]    wb_data,

// outputs
output [31:0] rs1_data,
output [31:0] rs2_data);
```

2. In your project top, use a push button (including the synchronizer from the seq_logic project in class) to provide the synchronous reset for the design. Use the CLK_100 as your system clock for now.

3. Create a register file with 32 registers each 32-bits in width.

4. If reset is asserted, synchronously zero the contents of all 32 registers.

5. Add functionality to asynchronously output the rs1_data associated with the contents of the register with the index rs1_reg.

6. Add functionality to asynchronously output the rs2_data associated with the contents of the register with the index rs2_reg.

7. Add functionality to synchronously writeback (on the positive edge of the clock) the value in wb_data to the register with the index wb_reg if wb_enable is asserted. Ensure that register 0 always contains a value of zero and cannot be overwritten if it is implemented as a register.

8. Use switches, LEDs, and the ILA as you wish to verify that the registers can be written and read.

9. Demonstrate the register read and write operations to the TA and send your Verilog source .sv files (rv32i_regs and top) to the TA for credit.