COSE222 Computer Architecture
Final Project Interview
(20% credit towards final project)

Update your pipelined version of MIPS, so that it is able to execute the code in the zip file. It will repeatedly display 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9 on HEX0 with a certain time interval.

http://esca.korea.ac.kr/teaching/cose222_CA/milestones/final-interview.zip

You should probably follow the steps below;

1. Compile the code under Eclipse and generate the MIPS binary.

2. Open `testvec.dump` to see what kinds of MIPS instructions you should implement
    a. Figure out which instructions are already there in the single-cycle CPU and what are not there
3. Implement those new instructions to your single-cycle CPU
4. **Simulate with ModelSim** to debug and validate your design.
5. Synthesize the pipelined CPU with the `mif` file using Quartus-II.
6. Download the bitstream to the DE0 board to make sure that it displays the numbers.

7. Implement the pipelined version of the MIPS.
8. **Simulate with ModelSim** to debug and validate your design.
9. Synthesize the pipelined CPU with the `mif` file using Quartus-II.
10. Download the bitstream to the DE0 board to make sure that it displays the numbers.

`MIPS_System.v` is an overly simplified version of a computer system, consisting of MIPS CPU, Timer, and GPIO. Nevertheless, it teaches you almost everything you need to know about computer systems. Please prepare to answer the following questions before coming to the one-on-one interview

1. We have extensively used the GPIO to display numbers on 7 segments. Check out how it is done. (Hint: Is it a memory-mapped I/O or I/O-mapped I/O? why and how?)
2. Check how the pointer in the C code is converted to assembly code.
3. You can program the Timer (`TimerCounter.v`) in the system. Check out the Verilog code to see what the purpose of the Timer is and how you would use it.

I am going to ask about the Verilog code you have designed for the milestones, too. So prepare the interview accordingly.

What and How to submit:

- Create a (up to) 3-min video clip (with your smartphone or any other convenient means), showing
  - Your smiling face to camera
  - 7 segment output on DE0 board

  AND **verbally** explaining the followings:
  - **What instructions** you added to the CPU, and **how you figured out those instructions** to be added to the CPU
  - **CPU design change** (please elaborate this!)
  - **ModelSim simulation output**

- Upload **both** the video clip **and** zipped Verilog source to Blackboard