Brown University
School of Engineering
EN164 Design of Computing Systems
Professor Sherief Reda
HW 03. 100 points. Due Date: Tuesday, April 10th in class.

---

1. [25 points] Assume the following datapath latencies for the individual stages.

   IF: 250 ps
   ID: 350 ps
   EX: 150 ps
   MEM: 300 ps
   WB: 200 ps

   a. [5 points] What is the clock cycle time in a pipelined and non-pipelined processor?
   b. [5 points] What is the total latency of an LW instruction in a pipelined and non-pipelined processor? Ignore the overhead of the pipeline registers.
   c. [5 points] If we can split one stage of the pipelined datapath into two new stages, each with half the latency of the original stage, which stage would you split and what is the new clock cycle time of the processor?

   Assume that the instructions executed by the processor are broke down as follows:

   ALU: 45%
   BEQ: 20%
   LW: 20%
   SW: 15%

   d. [5 points] Assume that are no stalls or hazards, what is the utilization of the data memory?
   e. [5 points] Assuming there are no stalls or hazards, what is the utilization of the write-register port of the register file?

2. [20 points] In this exercise, we examine how the ISA affects pipeline design. Assume the following new instruction:

ADDM Rd, Rt + Offset(Rs) # $Rd = $Rt + Mem[Offset+$Rs]

   a. [5 points] Give an example of where this instruction might be useful and a sequence of existing MIPS instructions that are placed by this instruction.
   b. [5 points] What must be changed in the pipelined datapath to add this instruction to the MIPS ISA?
   c. [5 points] Which new control signals must be added to the pipeline in part (a)?
   d. [5 points] Does support for this instruction introduce any new hazards? Are stalls due to existing hazards made worse?

3. [20 points]

```
I1:        LW     R1, 0 (R1)
I2:        AND    R1, R1, R2
I3:        LW     R2, 0 (R1)
I4:        LW     R1, 0 (R3)
```

a. [5 points] Find all data dependences in this instruction sequence.

b. [5 points] Find all hazards in this instruction sequence for a 5-stage pipeline with and then without forwarding

c. [10 points] To reduce clock cycle time, we are considering a split of the MEM stage into two stages, where reading completes in the second MEM stage. Repeat parts (a) and (b) for this 6-stage pipeline

4. [15 points] The importance of having a good branch predictor depends on how often conditional branches are executed. Together with branch predictor accuracy, this will determine how much time is spent stalling due to mispredicted branches. In this exercise, assume that the breakdown of dynamic instructions into various instruction categories is as follows:

R-Type: 40% BEQ: 25% JMP: 5% LW: 25% SW: 5%

Also assume the following branch predictor accuracies:

Always-Taken: 45% Always-Not-taken: 55% 2-Bit: 85%

a. [5 points] Stall cycles due to mispredicted branches increase the CPI. What is the extra CPI due to mispredicted branches with the always-taken predictor? Assume that branch outcomes are determined in the EX stage, that there are no data hazards, and that the no delay slot are used.

b. [5 points] Repeat (a) for the "Always-not-taken" predictor.

c. [5 points] Repeat (a) for the 2-bit predictor.

5. [20 points] Assume the following repeating pattern (e.g., in a loop) of branch outcomes:

T, NT, T, T, NT

a. [5 points] What is the accuracy of always-take and always-not-taken predictors for this sequence of branch outcomes?

b. [5 points] What is the accuracy of the two-bit predictor for the first four branches in this pattern, assuming that the predictor starts off in (predict not taken) state (bottom-left dark-grey state in the lecture slides)?

c. [5 points] What is the accuracy of the two-bit predictor if this pattern is repeated forever?

d. [5 points] Design a predictor that would achieve a perfect accuracy if this pattern is repeated forever. Your predictor should be a sequential circuit with one output that provides a prediction (1 for taken, 0 for not taken) and no inputs other than the clock and the control signal that indicates that the instruction is a conditional branch.