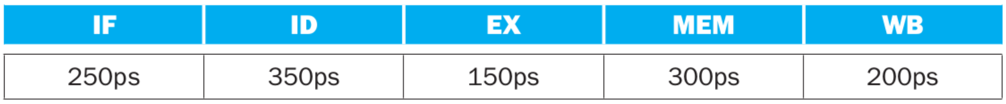
Cairo University	**Faculty of Engineering – Credit hour system**
Computer Engineering Department	**CMP N 301 – Computer Architecture**

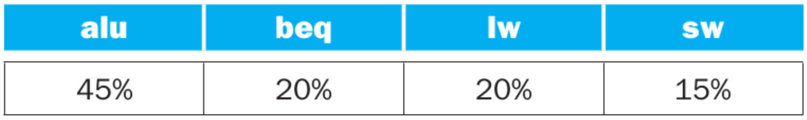
Sheet (2)
Chapter 4

1) [4.8] In this exercise, we examine how pipelining affects the clock cycle time of the processor. Problems in this exercise assume that individual stages of the datapath have the following latencies:



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

Also, assume that instructions executed by the processor are broken down as follows:



alu	beq	lw	sw
45%	20%	20%	15%

4.8.1 What is the clock cycle time in a pipelined and non-pipelined processor?

4.8.2 What is the total latency of an LW instruction in a pipelined and non-pipelined processor?

4.8.3 If we can split one stage of the pipelined datapath into two new stages, each with half latency of the original stage, which stage would you split and what is the new clock cycle time of the processor?

4.8.4 Assuming there are no stall or hazards, what is the utilization of the data memory?

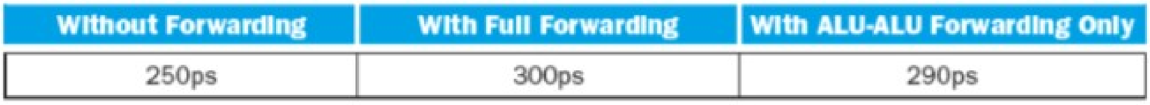
4.8.5 Assuming there are no stall or hazards, what is the utilization of the write-register port of the "Registers" unit?

4.8.6 Instead of a single-cycle organization, we can use a multi-cycle organization where each instruction takes multiple cycles but one instruction finishes before another is fetched. In this organization, an instruction only goes through stages it actually needs (e.g., ST only takes 4 cycles because it does not need the WB stage). Compare clock cycle times and execution times with single-cycle,multi-cycle, and pipelined organization.

2) [4.9] In this exercise, we examine how data dependences affect execution in the basic 5-stage pipeline described in Section 4.5. Problems in this exercise refer to the following sequence of instructions:

OR R1,R2,R3
OR R2,R1,R4
OR R1,R1,R2

Also, assume the following cycle times for each of the options related to forwarding:



Without Forwarding	With Full Forwarding	With ALU-ALU Forwarding Only
250ps	300ps	290ps

4.9.1 Indicate dependences and their type.

4.9.2 Assume there is no forwarding in this pipelined processor. Indicate hazards and add NOP instructions to eliminate them.

4.9.3 Assume there is full forwarding. Indicate hazards and add NOP instructions to eliminate them.

4.9.4 What is the total execution time of this instruction sequence without forwarding and with full forwarding? What is the speedup achieved by adding full forwarding to a pipeline that had no forwarding?

4.9.5 Add NOP instructions to this code to eliminate hazards if there is ALU-ALU forwarding only (no forwarding from the MEM to the EX stage).

4.9.6 What is the total execution time of this instruction sequence with only ALU-ALU forwarding? What is the speedup over a no-forwarding pipeline?