CMPE 200 HW#1

Name: Akshata Deo
SJSU ID: 012565761

---

1. **(a).** IPC = clock rate / CPI

   P1 = 3GHz/1.5 = $2*10^9$ instructions per second

   P2 = 4GHz/2 = $2*10^9$ instructions per second

   P3 = 5GHz/2.5 = $2*10^9$ instructions per second

   So, all three processors are same in terms of performance.

   (b). Now we want 2x speedup on all three processors. For P1 CPI increases by 10%. Therefore-

   Execution $Time_{new}$/Execution $Time_{old}$ = (I * (1.1 * CPI) * Clock cyle $time_{new}$) / (I * CPI * Clock cyle $time_{old}$)

   = (1.1 * Clock $rate_{old}$) / Clock $rate_{new}$ = 0.5

   Clock $rate_{new}$ = (1.1/0.5) * Clock $rate_{old}$ = (1.1/0.5) * 3GHz

   Clock $rate_{new}$ for P1 is 6.6 GHz

   Similarly, for P2-

   Clock $rate_{new}$ = (1.2/0.5) * Clock $rate_{old}$ = (1.2/0.5) * 4GHz

   Clock $rate_{new}$ for P2 is 9.6 GHz

   Similarly, for P3-

   Clock $rate_{new}$ = (1.3/0.5) * Clock $rate_{old}$ = (1.3/0.5) * 5GHz

   Clock $rate_{new}$ for P3 is 13 GHz

---

2. (a).

	Stack based	**Memory to memory**	**Register based**

Code Implementation	Push D
	Push A
	Sub
	Pop C
	Push B
	Push C
	Add
	Pop B
		SUB C, D, A
		ADD B, B, C
			Load R1, D
			Load R2, A
			Sub R3, R1, R2
			Store R3, C
			Load R4, B
			Add R4, R4, R3
			Store R4, B

Code Size	40 bytes	26 bytes	49 bytes
Memory Access	6	6	5 (and 11 register access)

(b). We know that code size = # instructions × size of each instruction
Size of each instruction = 1 byte*operations + 4 bytes*addresses + 2 bytes*registers.

Stack based size = 8 * (1 + 4) = 40 bytes
Memory to memory based size = 2 * (1 + 4*3) = 26 bytes
Register based size = 7 * (1 + 3*2) = **or** 7 * (1 + 2*1 + 4*1) = 49 bytes

Memory access for stack based is 6.
Memory access for memory to memory based is 6.
Memory access for register based is 5 (and 11 registers access).

(c). Stack has moderate code size and high memory access. Memory to memory based ISA has shortest code size and high memory traffic. Register based ISA has highest code size (since just 1 operation per instruction is getting performed) and low memory access because many operations get performed on registers instead of memory.

---

3. The critical path of sub instruction is I-Mem→Regs Rd→MUX→ALU→MUX→Regs Wr = 40 + 80 + 20 + 100 + 20 + 60 = 320 ps. Note that the RegDst latency can be overlapped with the two operand calculation (the destination register is only needed when the computation results come out).
   The critical path of lw instruction is I-Mem→Regs Rd→ALU→D-Mem→MUX→Regs Wr = 40 + 80 + 100 + 200 + 20 + 60 = 500 ps. Note that the path through the sign-extend and ALUSrc can be overlapped with register file access.
   Thus, the execution time of lw, 500ps, which is the longest will be set as clock period.

---

4. (a). Clock latency for single cycle processor: There is no pipelining in this, so the cycle-time has to allow an instruction to go through all the 5 stages each cycle. Therefore:
   Clock latency for single cycle processor = 100ps + 120ps + 220ps + 300ps + 120ps = 860ps

   (c). In single-cycle processor, 860ps (1 clock cycle).
