## Homework 5 – Performance and Pipelining

**Out:** 3.17.21
**Due:** 3.24.21

1. [Performance]
   Consider two different implementations of the same instruction set architecture. The instructions can be divided into four classes according to their CPI (class A, B, C, and D). P1 with clock rate of 2.5 GHz and CPIs of 1, 2, 3, and 3, and P2 with clock rate of 3 GHz and CPIs of 2, 2, 2, and 2.

   Given a program with a dynamic instruction count of 1.0E6 instructions divided into classes as follows: 10% class A, 20% class B, 50% class C, and 20% class D, which implementation is faster:
   a. What is the global CPI for each implementation?
   b. Find the clock cycles required in both cases.

2. [Performance]
   Compilers can have a profound impact on the performance of an application. Assume that for a program, compiler A results in a dynamic instruction count of 1.0E9 and has a resulting CPU execution time of 1.1 sec, while compiler B results in a dynamic instruction count of 1.2E9 and a resulting CPU execution time of 1.5 sec.
   a. Find the average CPI for each program given that the processor has a clock cycle time of 1 ns.
   b. Assume the compiled programs run on two different processors. If the execution times on the two processors are the same, how much faster is the clock of the processor running compiler A's code versus the clock of the processor running compiler B's code?
   c. A new compiler is developed that uses only 6.0E8 instructions and has an average CPI of 1.1. What is the speedup of using this new compiler versus using compiler A or B on the original processor?

3. [Performance]
   Utilization of a subset of the performance equation as a performance metric is considered a pitfall. To illustrate this, consider the following processors. P1 has a clock rate of 4 GHz, average CPI of 0.9, and requires the execution of 5.0E9 instructions. P2 has a clock rate of 3 GHz, an average CPI of 0.75, and requires the execution of 1.0E9 instructions.
   a. One usual fallacy is to consider the computer with the largest clock rate as having the largest performance. Check if this is true for P1 and P2.
   b. Another fallacy is to consider that the processor executing the largest number of instructions will need a larger CPU time. Considering processor P1 is executing a sequence of 1.0E9 instructions and that the CPI of processors P1 and P2 do not change, determine the number of instructions that P2 can execute in the same time that P1 needs to execute 1.0E9 instructions.

4. [Pipelining and Data Hazards]
   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 – 250ps, ID – 350ps, EX – 150 ps, MEM – 300ps, WB – 200ps.
   Also, assume that instructions executed by the processor are broken down as follows: ALU – 45%, beq – 20%, lw – 20%, sw – 15%
   1. What is the clock cycle time in a pipelined and non-pipelined processor?
   2. What is the total latency of an LW instruction in a pipelined and non-pipelined processor?
   3. 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?
   4. Assuming there are no stalls or hazards, what is the utilization of the data memory?
   5. Assuming there are no stalls or hazards, what is the utilization of the write-register port of the “Registers” unit (register file)?
   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 organizations.