1. Exercise 1.6 (page 55): 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 a clock rate of 2.5 GHz and CPIs of 1, 2, 3, and 3, and P2 with a clock rate of 3 GHz and CPIs of 2, 2, 2, and 2.

Given a program with a dynamic instruction count of $1.0 \times 10^6$ instructions divided into classes as follows: 10% class A, 20% class B, 50% class C, and 20% class D, which implementation is faster?

**P1: CPU CLOCK CYCLES**

\[(1 \times 10^5) + (2 \times 2 \times 10^5) + (3 \times 5 \times 10^5) + (3 \times 2 \times 10^5) = 2.6 \times 10^6\]

**P1: CPU TIME**

\[
\frac{2.6 \times 10^6}{2.5 \times \text{GHz}} = 1.04 \text{ ms}
\]

**P2: CPU CLOCK CYCLES**

\[(2 \times 10^5) + (2 \times 2 \times 10^5) + (2 \times 5 \times 10^5) + (2 \times 2 \times 10^5) = 2 \times 10^6\]

**P2: CPU TIME**

\[
\frac{2 \times 10^6}{3 \times \text{GHz}} = 666.67 \text{ ns}
\]

P2 is faster!

(a) What is the global CPI for each implementation?

\[
\text{CPI} = \frac{\text{CPU time}}{\text{Dynamic instruction count}}
\]

**P1: CPI**

\[
\frac{2.6 \times 10^6}{10^6} = 2.6
\]
(b) Find the clock cycles required in both cases.

**P1: CPU CLOCK CYCLES**

\[
2.6 \times 10^6
\]

**P2: CPU CLOCK CYCLES**

\[
2 \times 10^6
\]

2. **Exercise 1.7 (page 56):** 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 an execution time of 1.1 s, while compiler B results in a dynamic instruction count of 1.2E9 and an execution time of 1.5 s.

(a) Find the average CPI for each program given that the processor has a clock cycle time of 1 ns.

**A: CPI**

\[
\frac{1.1}{10^{-9} \times 10^9} = 1.1
\]

**B: CPI**

\[
\frac{1.5}{10^{-9} \times 1.2 \times 10^9} = 1.25
\]

(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?

**P1: CLOCK CYCLE TIME**

\[
\frac{10^9 \times 1.1}{10^9 \times 1.1}
\]

**P2: CLOCK CYCLE TIME**

\[
\frac{1.2 \times 10^9 \times 1.25}{1.2 \times 10^9 \times 1.25}
\]
\[
\frac{\text{CCT}_{P_1}}{\text{CCT}_{P_2}} = \frac{\frac{\text{EXECUTION TIME}}{10^9 \times 1.1}}{\frac{\text{EXECUTION TIME}}{1.2 \times 10^9 \times 1.25}} = \frac{1.2 \times 10^9 \times 1.25}{10^9 \times 1.1} \approx 1.36
\]

P1 is 36% slower!

(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?

**C : EXECUTION TIME**

\[6 \times 10^8 \times 1.1\text{ clock cycles}\]

**A : EXECUTION TIME**

\[10^9 \times 1.1\text{ clock cycles}\]

C is \(\frac{10}{6} \approx 1.67\) times faster than A!

**B : EXECUTION TIME**

\[1.2 \times 10^9 \times 1.25\text{ clock cycles}\]

C is \(\frac{15}{6} \approx 2.27\) times faster than B!

3. **Exercise 1.9 (page 56)**: Assume for arithmetic, load/store, and branch instructions, a processor has CPIs of 1, 12, and 5, respectively. Also assume that on a single processor a program requires the execution of 2.56E9 arithmetic instructions, 1.28E9 load/store instructions, and 256 million branch instructions. Assume that each processor has a 2 GHz clock frequency. Assume that, as the program is parallelized to run over multiple cores, the number of arithmetic and load/store instructions per processor is divided by 0.7 x p (where p is the number of processors) but the number of branch instructions per processor remains the same.

(a) Find the total execution time for this program on 1, 2, 4, and 8 processors, and show the relative speedup of the 2, 4, and 8 processor result relative to the single processor result.

\[
\text{EXECUTION TIME} = \frac{\text{CLOCK CYCLES}}{\text{CLOCK RATE}}
\]
1 PROCESSOR :
\[
\frac{(1 \times 2.56 \times 10^9) + (12 \times 1.28 \times 10^9) + (5 \times 2.56 \times 10^8)}{2\text{GHz}} = 9.6 \text{ s}
\]

2 PROCESSORS :
\[
\frac{(12 \times 1.28 \times 10^9) + (5 \times 2.56 \times 10^8)}{0.7 \times 2 \times 2\text{GHz}} = 7.04 \text{ s}
\]
Speed-up is 1.36

4 PROCESSORS :
\[
\frac{(1 \times 2.56 \times 10^9) + (12 \times 1.28 \times 10^9) + (5 \times 2.56 \times 10^8)}{0.7 \times 4 \times 2\text{GHz}} = 3.84 \text{ s}
\]
Speed-up is 2.5

8 PROCESSORS :
\[
\frac{(1 \times 2.56 \times 10^9) + (12 \times 1.28 \times 10^9) + (5 \times 2.56 \times 10^8)}{0.7 \times 8 \times 2\text{GHz}} = 2.24 \text{ s}
\]
Speed-up is 4.29

(b) If the CPI of the arithmetic instructions was doubled, what would the impact be on the execution time of the program on 1, 2, 4, or 8 processors?

1 PROCESSOR :
\[
9.6 + \frac{(1 \times 2.56 \times 10^9)}{2\text{GHz}} = 10.88 \text{ s}
\]
Slow-down is 1.13

2 PROCESSORS :
\[
7.04 + \frac{(1 \times 2.56 \times 10^9)}{0.7 \times 2 \times 2\text{GHz}} = 7.95 \text{ s}
\]
Slow-down is 1.13

4 PROCESSORS :
\[
3.84 + \frac{(1 \times 2.56 \times 10^9)}{0.7 \times 4 \times 2\text{GHz}} = 4.30 \text{ s}
\]
Slow-down is 1.12

8 PROCESSORS :
\[
7.04 + \frac{(1 \times 2.56 \times 10^9)}{0.7 \times 8 \times 2\text{GHz}} = 2.47 \text{ s}
\]
Slow-down is 1.1
(c) To what should the CPI of load/store instructions be reduced in order for a single processor to match the performance of four processors using the original CPI values?

\[(1 \times 2.56 \times 10^9) + (X \times 1.28 \times 10^9) + (5 \times 2.56 \times 10^8) = \\
\frac{(1 \times 2.56 \times 10^9) + (X \times 1.28 \times 10^9)}{0.7 \times 4} + (5 \times 2.56 \times 10^8)\]

\[X = \frac{3.84 \times 10^9}{1.28 \times 10^9} = 3\]

4. **Exercise 1.13 (page 58)**: Another pitfall cited in Section 1.10 is expecting to improve the overall performance of a computer by improving only one aspect of the computer. Consider a computer running a program that requires 250 s, with 70 s spent executing FP instructions, 85 s executing L/S instructions, and 40 s spent executing branch instructions.

(a) By how much is the total time reduced if the time for FP operations is reduced by 20%?

\[0.8 \times (\text{EXECUTION TIME}_{\text{FP}} + \text{EXECUTION TIME}_{\text{REST}}) = 0.8 \times 70 + (250 - 70) = 236\]

5.6% reduced

(b) By how much is the time for INT operations reduced if the total time is reduced by 20%?

\[200 = X \times (\text{EXECUTION TIME}_{\text{INT}} + \text{EXECUTION TIME}_{\text{REST}}) = X \times (250 - 70 - 85 - 40) + (70 + 85 + 40)\]

9% \approx X

91% reduced

(c) Can the total time can be reduced by 20% by reducing only the time for branch instructions?

\[200 = X \times (\text{EXECUTION TIME}_{\text{BR}} + \text{EXECUTION TIME}_{\text{REST}}) = X \times 40 + (250 - 40)\]

\[-10 \neq X \times 40\]

5. **Exercise 1.14 (page 59)**: Assume a program requires the execution of 50 × 10^6 FP instructions, 110 × 10^6 INT instructions, 80 × 10^6 L/S instructions, and 16 × 10^6 branch instructions. The CPI for each type of instruction is 1, 1, 4, and 2, respectively. Assume that the processor has a 2 GHz clock rate.
(a) By how much must we improve the CPI of FP instructions if we want the program to run two times faster?

\[
\text{EXECUTION TIME} = \frac{\text{CLOCK CYCLES}}{\text{CLOCK RATE}} = \frac{(50 \times 10^6 \times 1) + (110 \times 10^6 \times 1) + (80 \times 10^6 \times 4) + (16 \times 10^6 \times 2)}{2\text{GHz}} = 256 \text{ ms}
\]

\[
128 \text{ ms} = \frac{(50 \times 10^6 \times X) + (110 \times 10^6 \times 1) + (80 \times 10^6 \times 4) + (16 \times 10^6 \times 2)}{2\text{GHz}} - 4.12 \neq X
\]

(b) By how much must we improve the CPI of L/S instructions if we want the program to run two times faster?

\[
128 \text{ ms} = \frac{(50 \times 10^6 \times 1) + (110 \times 10^6 \times 1) + (80 \times 10^6 \times X) + (16 \times 10^6 \times 2)}{2\text{GHz}}
\]

\[
0.8 = X
\]

(c) By how much is the execution time of the program improved if the CPI of INT and FP instructions is reduced by 40% and the CPI of L/S and Branch is reduced by 30%?

\[
\text{EXECUTION TIME} = \frac{\text{CLOCK CYCLES}}{\text{CLOCK RATE}} = \frac{(50 \times 10^6 \times 0.6) + (110 \times 10^6 \times 0.6) + (80 \times 10^6 \times 2.8) + (16 \times 10^6 \times 1.4)}{2\text{GHz}} = 171.2 \text{ ms}
\]

Speed-up 1.49