Problem 1.6 (a)

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 1.0E6 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?

$$Weigted\ CPI = \sum_{i=1}^{n} CPI_i \times \frac{Instruction\ Count_i}{Instruction\ Count}$$

$$P_1\ CPI = \left[1 \times \left(\frac{.1 \times 1.0e6}{1.0e6}\right)\right] + \left[2 \times \left(\frac{.2 \times 1.0e6}{1.0e6}\right)\right] + \left[3 \times \left(\frac{.5 \times 1.0e6}{1.0e6}\right)\right] + \left[3 \times \left(\frac{.2 \times 1.0e6}{1.0e6}\right)\right] = 2.6$$

$$P_2\ CPI = \left[2 \times \left(\frac{.1 \times 1.0e6}{1.0e6}\right)\right] + \left[2 \times \left(\frac{.2 \times 1.0e6}{1.0e6}\right)\right] + \left[2 \times \left(\frac{.5 \times 1.0e6}{1.0e6}\right)\right] + \left[2 \times \left(\frac{.2 \times 1.0e6}{1.0e6}\right)\right] = 2$$

Problem 1.6 (b)

b. Find the clock cycles required in both cases.

$$Clock\ Cycles = \sum_{i=1}^{n} CPI_i \times Instruction\ Count_i$$

$$P_1\ Clock\ Cycles = [1 \times (.1 \times 1.0e6)] + [2 \times (.2 \times 1.0e6)] + [3 \times (.5 \times 1.0e6)] + [3 \times (.2 \times 1.0e6)] = 2.6e6$$

$$P_2\ Clock\ Cycles = [2 \times (.1 \times 1.0e6)] + [2 \times (.2 \times 1.0e6)] + [2 \times (.5 \times 1.0e6)] + [2 \times (.2 \times 1.0e6)] = 2.0e6$$

P2 is thus the faster implementation. It has fewer total clock cycles to get through and a faster clock speed as well.

Problem 1.9.1

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.

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.

One Processor:

$$Time = \frac{CPI \times Instruction\ Count}{Clock\ Rate}$$

$$Time = \frac{(1 \times 2.56e9) + (12 \times 1.28e9) + (5 \times 2.56e8)}{2e9} = 9.6\ secs$$

Two Processors:

$$Time = \frac{(\frac{1 \times 2.56e9}{.7 \times 2}) + (\frac{12 \times 1.28e9}{.7 \times 2}) + (5 \times 2.56e8)}{2e9} = 7.04\ secs$$

Four Processors:

$$Time = \frac{(\frac{1 \times 2.56e9}{.7 \times 4}) + (\frac{12 \times 1.28e9}{.7 \times 4}) + (5 \times 2.56e8)}{2e9} = 3.84\ secs$$

Eight Processors:

$$Time = \frac{(\frac{1 \times 2.56e9}{.7 \times 8}) + (\frac{12 \times 1.28e9}{.7 \times 8}) + (5 \times 2.56e8)}{2e9} = 2.24\ secs$$

Performance Comparison:

$$\frac{Single\ Processor\ Time}{Two\ Processors\ Time} = \frac{9.6}{7.04} \approx 1.36\ times\ improvement$$

$$\frac{Single\ Processor\ Time}{Four\ Processors\ Time} = \frac{9.6}{3.84} = 2.5\ times\ improvement$$

$$\frac{Single\ Processor\ Time}{Eight\ Processors\ Time} = \frac{9.6}{2.24} \approx 4.29\ times\ improvement$$

Problem 1.9.2

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?

One Processor:

$$Time = \frac{(2 \times 2.56e9) + (12 \times 1.28e9) + (5 \times 2.56e8)}{2e9} \approx 10.88\ secs$$

Two Processors:

$$Time = \frac{(\frac{2 \times 2.56e9}{.7 \times 2}) + (\frac{12 \times 1.28e9}{.7 \times 2}) + (5 \times 2.56e8)}{2e9} \approx 7.95\ secs$$

Four Processors:

$$Time = \frac{(\frac{2 \times 2.56e9}{.7 \times 4}) + (\frac{12 \times 1.28e9}{.7 \times 4}) + (5 \times 2.56e8)}{2e9} \approx 4.30\ secs$$

Eight Processors:

$$Time = \frac{(\frac{2 \times 2.56e9}{.7 \times 8}) + (\frac{12 \times 1.28e9}{.7 \times 8}) + (5 \times 2.56e8)}{2e9} \approx 2.47\ secs$$

Problem 1.9.3

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?

Set the performance calculation for a single core equal to the four core time and solve for the load/store CPI:

$$\frac{(1 \times 2.56e9) + (X \times 1.28e9) + (5 \times 2.56e8)}{2e9} = 3.84$$

$$X = \frac{(3.84 \times 2e9) - [(5 \times 2.56e8) + (1 \times 2.56e9)]}{1.28e9} = 3$$

The CPI of the load/store function must be 3.

Problem 1.13.1

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 executed L/S instructions, and 40 s spent executing branch instructions.

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

Time spent on FP operations = 70 secs.
Time spent on entire program = 250 secs.
Time spent on non-FP operations = 250 – 70 = 180 secs.

$$180 + [70 - (70 \times .2)] = 236\ secs.$$

Total time is reduced by 14 seconds to 236 seconds.

Problem 1.13.2

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

Total Time = 250 seconds
Assuming the remaining 55 seconds that are not taken up by FP instructions, L/S instructions, and branch instructions are taken up by int operations.

20% reduction of 250 seconds = 250 – (250 * .2) = 200.

If all operations are reduced by 20%, then int operations are reduced by 55 - [55 – (55 * .2)] = 11 seconds from 55 seconds to 44 seconds of time spent executing.

If it's just the total time that is reduced by 20%, then, since 85 + 70 + 40 = 195, int operations would only have 5 seconds to complete.

(This was a vague question).

Problem 1.13.3

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

Time spent total = 250 seconds.
Time spent on branch instructions = 40 seconds.
Time spent on non-branch instructions = 210 seconds.
Total time after 20% reduction = 200 seconds.

It's impossible to reduce the total time by 20% by only improving the execution time of branch instructions. There are always going to be 210 seconds of time devoted to executing non branch instructions.

Problem 1.14.1

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.

By how much must we improve the CPI of FP instructions if we want the program to run two times faster?

$$Cycles = 10^6 \times [(50 \times 1) + (110 \times 1) + (80 \times 4) + (16 \times 2)] = 5.12e8$$

If we cut the number of cycles in half, since the clock speed is the same in both cases, the program will speed up and become twice as fast. This means that we just have to solve the following equation:

$$\frac{5.12e8}{2} = 2.56e8 = 10^6 \times [(50 \times \boldsymbol{X}) + (110 \times 1) + (80 \times 4) + (16 \times 2)]$$

$$\frac{2.56e8}{10^6} = 256 = [(50 \times \boldsymbol{X}) + (110 \times 1) + (80 \times 4) + (16 \times 2)]$$

From here it becomes obvious that it is impossible to improve the program to run twice as fast by solely improving FP instructions. Because of the additions of constants like 110, 80, and 16 there are no positive values of X that would fit.

Problem 1.14.2

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

Following the logic of the previous question, we can solve for the CPI value that halves the number of cycles required.

$$\frac{5.12e8}{2} = 2.56e8 = 10^6 \times [(50 \times 1) \ + (110 \times 1) + (80 \times \boldsymbol{X}) \ + (16 \times 2)]$$

$$\frac{2.56e8}{10^6} = 256 = [(50 \times 1) \ + (110 \times 1) + (80 \times \boldsymbol{X}) \ + (16 \times 2)]$$

$$X = \frac{256 - [(50 \times 1) \ + (110 \times 1) + (16 \times 2)]}{80} = \ .8$$

We get a numerical answer of .8, so technically if the CPI of load instructions could be reduced to .8, the program could be run twice as fast. This is still impossible however, as every instruction will always take at least 1 cycle to complete.

Problem 1.14.3

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%?

$$Time_{Original} = \frac{10^6 \times [(50 \times 1) \; + (110 \times 1) + (80 \times 4) \; + (16 \times 2)]}{2 \times 10^9} = \; .256 \; sec$$

$$Time_{Improved} = \frac{10^6 \times [(50 \times 1 \times .4) \; + (110 \times 1 \times .4) + (80 \times 4 \times .3) \; + (16 \times 2 \times .3)]}{2 \times 10^9} = \; .0848 \; sec$$

.256 - .0848 = .1712 sec

.256 / .0848 = 3.01886

The new program runs .1712 seconds faster (about 3 times as fast).