Ch. 1 (5, 6, 7, 13)

Jason Lee
Y04640158

CSM151B Homework #1

(1.5) P1 => 3 GHz clk rate, 1.5 CPI
P2 => 2.5, 1.0
P3 => 4, 2.2

(a) (P1) => $\left(\frac{3\times10^9 \text{ cyc}}{s}\right)\left(\frac{1 \text{ instr}}{1.5 \text{ cyc}}\right) = 2\times10^9 \text{ instr/s}$

(P2) => $\left(\frac{2.5\times10^9 \text{ cyc}}{s}\right)\left(\frac{1 \text{ instr}}{1 \text{ cyc}}\right) = 2.5\times10^9 \text{ instr/s}$

(P3) => $\left(\frac{4\times10^9 \text{ cyc}}{s}\right)\left(\frac{1 \text{ instr}}{2.2 \text{ cyc}}\right) = 1.82\times10^9 \text{ instr/s}$

=> P2 has highest performance

(b) # of cycles

(P1) => $\left(\frac{3\times10^9 \text{ cyc}}{s}\right)\left(\frac{10 \text{ s}}{1}\right) = 3\times10^{10}$ cycles

(P2) => $\left(\frac{2.5\times10^9 \text{ cyc}}{s}\right)\left(\frac{10 \text{ s}}{1}\right) = 2.5\times10^{10}$ cycles

(P3) => $\left(\frac{4\times10^9 \text{ cyc}}{s}\right)\left(\frac{10 \text{ s}}{1}\right) = 4\times10^{10}$ cycles

# of instr

(P1) => $\left(\frac{3\times10^9 \text{ cyc}}{s}\right)\left(\frac{10 \text{ s}}{1}\right)\left(\frac{\text{instr}}{1.5 \text{ cyc}}\right) = 2\times10^{10}$ instr

(P2) => $\left(\frac{2.5\times10^9 \text{ cyc}}{s}\right)\left(\frac{10 \text{ s}}{1}\right)\left(\frac{\text{instr}}{1 \text{ cyc}}\right) = 2.5\times10^{10}$ instr

(P3) => $\left(\frac{4\times10^9 \text{ cyc}}{s}\right)\left(\frac{10 \text{ s}}{1}\right)\left(\frac{\text{instr}}{2.2 \text{ cyc}}\right) = 1.82\times10^{10}$ instr

(c) Exec Time = (# instr)(CPI)(Clk Cyc Time)

=> $\frac{\text{Exec Time new}}{\text{Exec Time old}} = \frac{(\#\text{ instr})(1.2\text{ CPI})(\text{Clk Cyc Time new})}{(\#\text{ instr})(\text{CPI})(\text{Clk Cyc Time old})} = \frac{1.2\text{ Clk rate old}}{\text{Clk rate new}} = 0.7$

= Clk rate new = $\left(\frac{1.2}{0.7}\right)$ Clk rate old => Clk rate new = 1.71 Clk rate old

= 71% increase

(1.6) P1 ⇒ 2.5 GHz [1, 2, 3, 3 CPIs]
P2 ⇒ 3 GHz [2, 2, 2, 2 CPIs]

- Dynamic Inst Count ⇒ 1×10^6 instr
- Classes ⇒ [10% A, 20% B, 50% C, 20% D]

Which implementation is faster?

(P1) $1\ s = 2.5 \times 10^9$ instr

$1\ cyc = \frac{1}{2.5 \times 10^9} = 4 \times 10^{-10}\ s = 0.4 \times 10^{-9}\ s = 400\ ps$

(P2) $1\ s = 3 \times 10^9$ instr

$1\ cyc = \frac{1}{3 \times 10^9} = 3.33 \times 10^{-10}\ s = 0.333 \times 10^{-9}\ s = 333\ ps$

Total CPU Time

(P1) ⇒ $[[(0.1)(1)(400\ ps)] + [(0.2)(1)(400\ ps)] + [(0.5)(3)(400\ ps)] + [(0.2)(3)(400\ ps)]](1 \times 10^6\ instr)$
$= (40 + 160 + 600 + 240)(1 \times 10^6) = 1\,040\,000\,000\ ps = 1.04\ ms$

(P2) ⇒ $[[(0.1)(2)(333\ ps)] + [(0.2)(2)(333\ ps)] + [(0.5)(2)(333\ ps)] + [(0.2)(2)(333\ ps)]](1 \times 10^6\ instr)$
$= (66.6 + 133.2 + 333 + 133.2)(1 \times 10^6) = 666{,}000{,}000\ ps = 0.666\ ms$

- Thus, $\boxed{\text{P2 is faster}}$

(a) Class A = 1×10^5 instr
B = 2×10^5
C = 5×10^5
D = 2×10^5

Time = $\frac{(\#\ instr)(CPI)}{Clk\ Rate}$

(P1) ⇒ $\frac{[(1 \times 10^5)(1)] + [(2 \times 10^5)(2)] + [(5 \times 10^5)(3)] + [(2 \times 10^5)(3)]}{2.5 \times 10^9} = 10.4 \times 10^{-4}\ s$

(P2) ⇒ $[(1 \times 10^5)(2)] + [(2 \times 10^5)(2)] + [(5 \times 10^5)(2)] + [(2 \times 10^5)(2)] = 6.67 \times 10^{-4}\ s$

(P1) ⇒ Global CPI = $\frac{(10.4 \times 10^{-4})(2.5 \times 10^9)}{1 \times 10^6} = \boxed{2.6}$

(P2) ⇒ Global CPI = $\frac{(6.67 \times 10^{-4})(3 \times 10^9)}{1 \times 10^6} = \boxed{2.0}$

(cont)

1.6 (cont) (b) Clock Cycles

(P1) $(1\times10^5)(1) + (2\times10^5)(2) + (5\times10^5)(3) + (2\times10^5)(3) = \boxed{2.6\times10^6 \text{ clk cycles}}$

(P2) $(1\times10^5)(2) + (2\times10^5)(2) + (5\times10^5)(2) + (2\times10^5)(2) = \boxed{2\times10^6 \text{ clk cycles}}$

1.7 Comp. A ⇒ Dynamic instr cnt $= 1\times10^9$, Exec Time $= 1.1s$
Comp. B ⇒ $= 1.2\times10^9$, $1.5s$

(a) Avg. CPI given proc. has clk cycle $= 1ns = 1\times10^{-9}s$

CPU Time = (# instr)(CPI)(Clk cycle Time)

(A) ⇒ $CPI = \frac{\text{CPU Time}}{(\#\text{ instr})(\text{clk cyc Time})} = \frac{1.1}{(1\times10^9)(1\times10^{-9})} = \boxed{1.1}$

(B) ⇒ $= \frac{1.5}{(1.2\times10^9)(1\times10^{-9})} = \boxed{1.25}$

(b) Exec Time = (# instr)(CPI)(Clk cyc time)

⇒ $\frac{Clk_A}{Clk_B} = \frac{(\#\text{ instr}_B)(CPI_B)}{(\#\text{ instr}_A)(CPI_A)} \Rightarrow \frac{(1.2\times10^9)(1.25)}{(1\times10^9)(1.1)} = 1.36$

- Thus, clk of proc 1 running A's code is 36% slower than for the clk of proc 2 running B's code

(c) Comp. C ⇒ 6×10^8 instr, CPI = 1.1

(A) $\frac{\text{Performance C}}{\text{Performance A}} = \frac{\text{Exec Time A}}{\text{Exec Time C}} \Rightarrow \frac{(1.1)(1\times10^9)}{(1.1)(6\times10^8)} = 1.67$

(B) $\frac{(1.25)(1.2\times10^9)}{(1.1)(6\times10^8)} = 2.27$

- Thus, compiler C's code is 1.67 times faster than A's code is 2.27 times faster than B's code

(1.13) Total Exec Time = 250 s

70 s =	FP
85 s =	L/S
40 s =	Branch
55 s =	INT

(1.13.1) • Reducing FP Time by 20% ⇒ (70 s)(0.8) = 56 s

• New Time ⇒ 250 s − 70 s + 56 s = 236 s

• Time saved ⇒ 250 s − 236 s = 14 s

• Percentage of total time reduced ⇒ $\left(\frac{14\,s}{250\,s}\right)(100\%) = \boxed{5.6\%}$

(1.13.2) • INT instr CPU Time = L/S instr Exec Time − 0.2 L/S instr Exec Time

= (85 s) − (0.2)(85 s) = 68 s

• New Total Exec Time = 70 s + 68 s + 40 s + 55 s = 233 s

• Time saved ⇒ 250 s − 233 s = 17 s

• Percentage of total time reduced ⇒ $\left(\frac{17\,s}{250\,s}\right)(100\%) = \boxed{6.8\%}$

(1.13.3) • New Total Time ⇒ Tot Time − 0.2 Tot Time

= (250 s) − (0.2)(250 s) = 200 s

• Suppose the branch instr time is reduced by 100% ⇒ 70 s + 85 s + 55 s = 210 s

• Thus, the total time can't be reduced by 20% by only reducing the branch instr time