04S1 COMP3211/9211 Computer Architecture
Tutorial 1 (Weeks 02 & 03) Solutions

Lih Wen Koh (lwkoh@cse)

September 14, 2004

Key: SRQ = Stallings, Review Question; SP = Stallings Problem; P = Patterson & Hennessy Exercise

Warmup Exercises

Q3. [P2.1] We wish to compare the performance of two different machines: M1 and M2. The following measurements have been make on these machines:

program	time on M1	time on M2
1	10 seconds	5 seconds
2	3 seconds	4 seconds

Which machine is faster for each program and by how much?

Recall, A is n times faster than B for program P if

$$\frac{CPU\ Performance_{\ A}}{CPU\ Performance_{\ B}} = \frac{Execution\ Time_{\ B}}{Execution\ Time_{\ A}} = n$$

For program 1, M2 is $\frac{10}{5}$ = 2 times faster than M1.

For program 2, M1 is $\frac{4}{3}$ = 1.33 times faster than M2.

Q4. [P2.2] Consider the two machines and programs in P2.1. The following additional measurements were made:

program	instructions executed on M1	instructions executed on M2
1	200 x 10^6	160 x 10^6

Find the instruction execution rate (instructions per second) for each machine when running program 1.

$$Instruction\ execution\ rate\ for\ M1\ = \frac{200 * 10^6\ instructions}{10\ secs} = 20\ MIPS$$

$$Instruction\ execution\ rate\ for\ M2\ = \frac{160 * 10^6\ instructions}{5\ secs} = 32\ MIPS$$

Q5. [P2.3] If the clock rates of machines M1 and M2 in P2.1 are 200MHz and 300MHz, respectively, find the clock cycles per instruction (CPI) for program 1 on both machines using the data in P2.1 and P2.2.

Recall

$$\begin{aligned} CPU\ time &= \frac{instructions}{program} * \frac{cycles}{instructions} * \frac{seconds}{cycles} \\ &= instruction\ count\ *\ CPI\ *\ clock\ period \\ So\ \ CPI &= \frac{CPU\ time}{instruction\ count\ *\ clock\ period} \\ &= \frac{CPU\ time\ *\ clock\ rate}{instruction\ count} \end{aligned}$$

For program 1,

$$CPI for M1 = \frac{10\ *\ 200 * 10^6}{200 * 10^6} = 10$$
$$CPI for M2 = \frac{5\ *\ 300 * 10^6}{160 * 10^6} = 9.375$$

Q6. [P2.4] Assuming the CPI for program 2 on each machine in Q3 is the same as the CPI for program 1 found in Q5, find the instruction count for program 2 running on each machine using the execution times from Q3.

From Q5, for program 1: CPI for M1 = 10; CPI for M2 = 9.375.
Given that program 2 has the same CPI for program 1 for both machines M1 and M2.
Also, recall

$$\begin{aligned} CPU\ time &= instruction\ count\ *\ CPI\ *\ clock\ period \\ So\ \ instruction\ count &= \frac{CPU\ time}{CPI\ *\ clock\ period} \\ &= \frac{CPU\ time\ *\ clock\ rate}{CPI} \end{aligned}$$

Thus for program 2,

$$instruction\ count\ on\ M1\ = \frac{3\ *\ 200 * 10^6}{10} = 60\ *\ 10^6$$
$$instruction\ count\ on\ M2\ = \frac{4\ *\ 300 * 10^6}{9.375} = 128\ *\ 10^6$$

Q7. [P2.5] Suppose that M1 in P2.1 costs $10,000 and M2 costs $15,000. If you needed to run the program 1 a large number of times (i.e., if you were concerned with throughput instead of response time), which machine would you buy in large quantities? Why?

Buy M2. M2 is faster than M1 by 2 times but cost of M2 is less than twice that of M1.

Discussion Questions

Q10. [P1.48] What is the approximate relationship between cost and die area? The approximate relationship can be described as

$$Cost = f((Die\ area)^x)$$

for some x. You don't have to determine f, but you can determine x by first writing

$$Dies\ per\ wafer = f((Die\ area)^y)$$

$$Yield = f((Die\ area)^z)$$

and then examining how these two equations impact the first one (you need to figure out what y and z are). What implications does this have for designers?

We can write

$$\begin{aligned} Cost\ per\ die &= \frac{Cost\ per\ wafer}{Dies\ per\ wafer * yield} \\ Dies\ per\ wafer &= \frac{Wafer\ area}{Die\ area} \\ Yield &= \frac{1}{(1 + Defect\ per\ area * Die\ area/2)^2} \end{aligned}$$

So

$$y = -1; \quad z = -2; \quad x = 3.$$

Q11. [P2.10] Consider two different implementations, M1 and M2, of the same instruction set. There are four classes of instructions (A, B, C, and D) in the instruction set.

M1 has a clock rate of 500 MHz. The average number of cycles for each instruction class on M1 is as follows:

class	CPI for this class
A	1
B	2
C	3
D	4

M2 has a clock rate of 750 MHz. The average number of cycles for each instruction class on M2 is as follows:

class	CPI for this class
A	2
B	2
C	4
D	4

Assume that peak performance is defined as the fastest rate that a machine can execute an instruction sequence chosen to maximize that rate.
What are the peak performances of M1 and M2 expressed as instructions per second?

$$Peak\ Performance = \frac{Cycles\ per\ second}{Cycles\ to\ execute\ fastest\ instructions}$$

M1:
Choose instructions from class A (lowest CPI).
Peak performance $= \frac{500*10^6}{1} = 500\ MIPS$

M2:
Choose instructions from either class A or B.
Peak performance $= \frac{750*10^6}{2} = 375\ MIPS$

Q12. [P2.13] Consider two different implementations, M1 and M2, of the same instruction set. There are three classes of instructions (A, B, and C) in the instruction set. M1 has a clock rate of 400 MHz, and M2 has clock rate of 200 MHz. The average number of cycles for each isntruction class on M1 and M2 is given in the following table:

Class	CPI on M1	CPI on M2	C1 Usage	C2 Usage	Third-party usage
A	4	2	30 %	30 %	50 %
B	6	4	50 %	20 %	30 %
C	8	3	20 %	50 %	20 %

The table also contains a summary of how three different compilers use the instructions set. C1 is a compiler produced by the makers of M1, C2 is a compiler produced by the makers of M2, and the other compiler is a

third- party product. Assume that each compiler uses the same number of instructions for a given program but that the instruction mix is as described in the table.

(a) Using C1 on both M1 and M2, how much faster can the makers of M1 claim that M1 is compared with M2?

Using C1 compiler,

$$\textit{CPI for M1} = 0.3 * 4 + 0.5 * 6 + 0.2 * 8 = 5.8$$
$$\textit{CPI for M2} = 0.3 * 2 + 0.5 * 4 + 0.2 * 3 = 3.2$$

Recall

$$CPU\ time\ =\ instruction\ count\ *\ CPI\ *\ clock\ period$$

Let I be the instruction count for a given program. Then

$$\frac{Performance\ of\ M2}{Performance\ of\ M1} = \frac{I\ *\ 3.2\ *\ (200\ *\ 10^6)^{-1}}{I\ *\ 5.8\ *\ (400\ *\ 10^6)^{-1}} = 1.103$$

M1 is faster than M2 by 1.103 times.

(b) Using C2 on both M2 and M1, how much faster can the makers of M2 claim that M2 is compared with M1?

Using C2 compiler,

$$\textit{CPI for M1} = 0.3 * 4 + 0.2 * 6 + 0.5 * 8 = 6.4$$
$$\textit{CPI for M2} = 0.3 * 2 + 0.2 * 4 + 0.5 * 3 = 2.9$$

Let I be the instruction count for a given program. Then

$$\frac{Performance\ of\ M1}{Performance\ of\ M2} = \frac{I\ *\ 6.4\ *\ (400\ *\ 10^6)^{-1}}{I\ *\ 2.9\ *\ (200\ *\ 10^6)^{-1}} = 1.103$$

M2 is faster than M1 by 1.103 times.

(c) If you purchase M1, which compiler would you use?

(d) If you purchase M2, which compiler would you use?

Using third party compiler,

$$\textit{CPI for M1} = 0.5 * 4 + 0.3 * 6 + 0.2 * 8 = 5.4$$
$$\textit{CPI for M2} = 0.5 * 2 + 0.3 * 4 + 0.2 * 3 = 2.8$$

Execution time for a program is directly proportional to the CPI for the program. Thus we would like to choose the compiler that gives the lowest

CPI for a particular machine. For both M1 and M2, the third party compiler gives the lowest CPI.

(e) Which machine would you purchase if we assume that all other criteria are identical, including costs?

Using third-party compiler,
Let I be the instruction count for a given program. Then

$$\frac{Performance\ of\ M2}{Performance\ of\ M1} = \frac{I\ *\ 2.8\ *\ (200\ *\ 10^6)^{-1}}{I\ *\ 5.4\ *\ (400\ *\ 10^6)^{-1}} = 1.037$$

M1 is faster than M2 by 1.037 times. So we purchase machine M1.

Q13. [P2.15] We are interested in two implementations of a machine, one with and one without special floating-point hardware.

Consider a program, P, with the following mix of operations:

floating-point multiply	10 %
floating-point add	15 %
floating-point divide	5 %
integer instructions	70 %

Machine MFP (Machine with Floating Point) has floating-point hardware and can therefore implement the floating-point operations directly. It requires the following number of clock cycles for each instruction class:

floating-point multiply	6
floating-point add	4
floating-point divide	20
integer instructions	2

Machine MNFP (Machine with No Floating Point) has no floating-point hardware and so must emulate the floating-point operations using integer instructions. The integer instructions all take 2 clock cycles each. The number of integer instructions needed to implement each of the floating-point operations is as follows:

floating-point multiply	30
floating-point add	20
floating-point divide	50

Both machines have a clock rate of 1000 MHz. Find the native MIPS ratings for both machines.

Recall

$$MIPS\ rating = \frac{Instruction\ count}{Execution\ time\ *\ 10^6} = \frac{Clock\ rate}{CPI\ *\ 10^6}$$

*CPI for MFP = 0.1*6 + 0.15*4 + 0.05*20 + 0.7*2 = 3.6*
CPI for MNFP = 2 (All floating-point instructions have been replaced by integer instructions.)

Thus,
MIPS rating for MFP $= \frac{1000}{3.6} = 278$

MIPS rating for MFP $= \frac{1000}{2} = 500$

Q14. [P3.14] When designing memory systems, it becomes useful to know the frequency of memory reads versus writes as well as the frequency of accesses for instructions versus data. Using the average instruction mix information for MIPS for the progrqam gcc, find the following:

Instruction	gcc (freq)
Arithmetic	48 %
Data transfer	33 %
Conditional branch	17 %
Jump	2 %

(a) The percentage of *all* memory accesses that are for data (vs. instructions).

Given that 33% of instructions are data accesses, so

$$\begin{aligned} \textit{Percentage of data memory accesses} &= \frac{\#\ \textit{Data accesses}}{\#\ \textit{Instructions} + \#\ \textit{Data accesses}} * 100\% \\ &= \frac{0.33I}{I+0.33I} * 100\% \\ &= 24.81\% \end{aligned}$$

(b) The percentage of *all* memory accesses that are reads (vs. writes). Assume that two-thirds of data transfers are loads.

Given that 33% of instructions are data accesses; and two thirds of data accesses are loads, so

$$\begin{aligned} \textit{Percentage of read memory accesses} &= \frac{\#\ \textit{Instructions} + \#\ \textit{Data loads}}{\#\ \textit{Instructions} + \#\ \textit{Data accesses}} * 100\% \\ &= \frac{I + 0.33I*\frac{2}{3}}{I + 0.33I} * 100\% \\ &= 91.73\% \end{aligned}$$

Q15. [P3.16] Suppose we have made the following measurements of average CPI for instructions:

Instruction	Average CPI	gcc (freq)	spice (freq)
Arithmetic	1.0 clock cycles	48 %	50 %
Data transfer	1.4 clock cycles	33 %	41 %
Conditional branch	1.7 clock cycles	17 %	8 %
Jump	1.2 clock cycles	2 %	1 %

Compute the effective CPI for MIPS. Average the instruction frequencies for gcc and spice to obtain the instruction mix.

*Effective CPI for gcc = 1.0*0.48 + 1.4*0.33 + 1.7*0.17 + 1.2*0.02 = 1.255*

*Effective CPI for spice = 1.0*0.5 + 1.4*0.41 + 1.7*0.08 + 1.2*0.01 = 1.222*