EE 457 HW 4
Performance
Redekopp • Puvvada

Name: _____

Due: See Website **Score: _____**

Please post any questions regarding HW problems on Piazza.

This homework contains some problems from the 3rd edition of "Computer Organization and Design" by Hennessy and Patterson. We reproduce the problems below.

1. **(3) [3rd Ed., Q4.1]** We wish to compare the performance of two different computers: M1 and M2. The following measurements have been made on these computers. Which computer is faster for each program and how many times as fast is it (i.e. what is the speedup)?

Program	Time on M1	Time on M2
1	**2.0 sec.**	**1.5 sec.**
2	**5.0 sec.**	**10.0 sec.**

2. **(4) [3rd Ed., Q4.2]** Consider the two computers and programs in [the previous problem]. The following additional measurements were made. Find the instruction execution rate (instructions per second) for each computer when running program 1

Program	Instructions Executed on M1	Instructions Executed on M2
1	**5×10^9**	**6×10^9**

3. **(4) [3rd Ed., Q4.17]** The clock rates of computers M1 and M2 in the previous exercises are 4GHz and 6GHz, respectively, find the clock cycles per instruction (CPI) for program 1 on both computers using the data [provided in Exercise 4.1 and 4.2].

4. **(4) [3rd Ed., Q4.18]** Assuming the CPI for program 2 on each computer in [Exercise 4.1] is the same as the CPI for program 1 found in [Exercise 4.17], find the instruction count for program 2 running on each computer using the execution times in [Exercise 4.1].

5. **(5) [3rd Ed., Q4.8]** Consider two different implementations, P1 and P2, of the same instruction set. There are five classes of instructions (A, B, C, D, and E) in the instruction set. P1 has a clock rate of 4 GHz. P2 has a clock rate of 6 GHz. The average number of cycles for each instruction class for P1 and P2 is as shown in the table below. Assume that peak performance is defined as the fastest rate that a computer can execute any instruction sequence. What are the peak performances of P1 and P2 expressed in instructions per second.

Class	CPI on P1	CPI on P2
A	1	2
B	2	2
C	3	2
D	4	4
E	3	4

6. **(2) [3rd Ed., Q4.11]** Consider program P, which runs on a 1GHz machine M in 10 seconds. An optimization is made to P, replacing all instances of multiplying a value by 4 (mult X, X, 4) with two instructions that set x = x + x twice. Call this new optimized program, P'. The CPI of a multiply instruction is 4, and the CPI of an 'add' is 1. After recompiling, the program now runs in 9 seconds on machine M. How many multiplies were replaced by the new compiler?

7. **(10) [3rd Ed., Q4.19]** Suppose we enhance a computer to make all floating-point instructions run five times faster. Let's look at how speedup behaves when we incorporate the faster floating-point hardware. If the execution time of some benchmark before the floating-point enhancement is 10 seconds, what will the speedup be if half of the 10 seconds was spent executing floating-point instructions?

8. **(10) [3rd Ed., Q4.20]** We are looking for a benchmark to show off the new floating-point unit described in Exercise 4.19, and we want the overall benchmark to show a speedup of 3. One benchmark we are considering runs for 100 seconds with the old floating-point hardware. How much of the initial execution time would floating-point instructions have to account for to show an overall speedup of 3 on this benchmark?

Taken From Spring 1994 Midterm

9. **(5)** A RISC processor has a MIPS rating of 80 MIPS and a CISC processor has a MIPS rating of 20 MIPS. Is it *fair* to say that the RISC processor is about four times faster than the CISC processor? Explain very briefly?

Taken From Summer 1994 Midterm

10. **(12)** Hardware multiplier in CISC processors. Processor #1 and #2 have the same instruction-set architecture and are clocked by the same frequency clock.

    Processor #1 has a hardware multiplier and takes 8 clocks to finish an integer multiplication. Processor #2 is the same as processor #1 but has an improved hardware multiplier which takes only 4 clocks to finish the integer multiplication.

    Consider the following two ***separate*** supplemental data.

    (i) Processor #1 spends 10% of its time in performing integer multiplication while executing typical benchmark programs.
    (ii) It is found that the frequency of occurrence of the integer multiplication instruction in the object code, generated by compiling typical benchmark programs, is 10%.

    Answer the question and provide work to support your answer.

    10.1. **(Yes / No)** Using supplemental data (i), but not (ii), is it possible to find the speedup of processor #2 over processor #1?
    10.2. **(Yes / No)** Using supplemental data(ii), but not (i), is it possible to find speed-up of processor #2 over processor #1?
    10.3. **(Yes / No)** Does supplemental data (i) imply supplemental data (ii)?
    10.4. **(Yes / No)** Does supplemental data (ii) imply supplemental data (i)?

Taken From Summer 1995 Midterm

11. **(21)** Two CPU manufacturers (XYZ and ABC) implemented the same ISA (instruction set architecture). They each have their own datapath and control unit (multi-cycle) designs. Let us assume that there are only three kinds of instructions (A, B, and C) here.

	XYZ	ABC
Clock Frequency	100 MHz	50 MHz
CPI A	4	2
CPI B	6	4
CPI C	8	3

11.1. **(3)** Consider each instruction type and find out which CPU executes that instruction faster:

(i)	**Instr. A**	a.) Both are same	b.) XYZ is faster	c.) ABC is faster
(ii)	**Instr. B**	a.) Both are same	b.) XYZ is faster	c.) ABC is faster
(iii)	**Instr. C**	a.) Both are same	b.) XYZ is faster	c.) ABC is faster

11.2. There is a standard third party compiler, and also there are special compilers produced by XYZ and ABC. The compilers use the A, B, and C instructions in the percentages shown below. Assume that the instruction counts of binaries produced by the three compilers are all the same. Notice that the special compilers are designed to make the competitor look relatively bad.

	Standard Compiler	Special for XYZ	Special for ABC
A	50%	30%	30%
B	30%	50%	20%
C	20%	20%	50%

11.2.1. **(4)** Using the binaries produced by its special compiler XYZ claims to be faster than ABC by a factor of _____.

11.2.2. **(4)** Using the binaries produced by its special compiler ABC claims to be faster than XYZ by a factor of _____.

11.2.3. **(4)** If binaries produced by the standard compiler are used who looks better: **(a)** XYZ or **(b)** ABC?

11.2.4. **(3)** If you buy the CPU from XYZ, which compiler would you use?
- **(i)** Standard compiler
- **(ii)** Special Compiler from XYZ
- **(iii)** Special Compiler from ABC

11.2.5. **(3)** If you buy the CPU from ABC, which compiler would you use?
- **(i)** Standard compiler
- **(ii)** Special Compiler from XYZ
- **(iii)** Special Compiler from ABC

Taken From Summer 2001 Midterm

12. **(22)** A standard benchmark, when compiled with a standard compiler used the four categories of instructions as tabulated below in Table 1.

	Frequency	CPI	Freq. x CPI
NOP	0%	2	0.0 x 2 = 0.0
A	40%	3	0.4 x 3 = 1.2
B	20%	4	0.2 x 4 = 0.8
C	40%	5	0.4 x 5 = 2.0

Table 1

(6) For the same benchmark, Mr. Cheat made the compiler to add NOP instructions (50% of the total as shown below) to fictitiously boost the performance. Complete table 2 below for Mr.Cheat.

	Frequency	CPI	Freq. x CPI
NOP	50%	2	0.5 x 2 = 1.0
A	___%	3	____ x 3 = ____
B	___%	4	____ x 4 = ____
C	___%	5	____ x 5 = ____

Table 2

12.1. **(4)** Calculate CPI for Table 1 = _____

12.2. **(4)** Calculate CPI for Table 2 = _____

12.3. **(2)** Are the above percentages based on the frequency of occurrence of the instructions (a.) static code or (b.) dynamic execution trace? _____

12.4. **(6)** By adding NOPS did the ***actual*** (real, not fictitious) performance go down or go up? (**DOWN / UP) _____**. By what factor? _____ If you do not have enough data, state what data is essential to calculate, make assumptions for that needed data and proceed. Show your calculations briefly.