

The historical perspective for this chapter provides a background for some of the key ideas presented in this opening chapter. Its purpose is to give you the human story behind the technological advances and to place achievements in their historical context. By understanding the past, you may be better able to understand the forces that will shape computing in the future. Each historical perspectives section on the CD ends with suggestions for further reading, which are also collected separately on the CD under the section “Further Reading.” The rest of Section 1.10 is found on the CD.

1.11

Exercises

Contributed by Javier Bruguera of Universidade de Santiago de Compostela

Most of the exercises in this edition are designed so that they feature a qualitative description supported by a table that provides alternative quantitative parameters. These parameters are needed to solve the questions that comprise the exercise. Individual questions can be solved using any or all of the parameters—you decide how many of the parameters should be considered for any given exercise question. For example, it is possible to say “complete Question 4.1.1 using the parameters given in row A of the table.” Alternately, instructors can customize these exercises to create novel solutions by replacing the given parameters with your own unique values.

The number of quantitative exercises varies from chapter to chapter and depends largely on the topics covered. More conventional exercises are provided where the quantitative approach does not fit.

The relative time ratings of exercises are shown in square brackets after each exercise number. On average, an exercise rated [10] will take you twice as long as one rated [5]. Sections of the text that should be read before attempting an exercise will be given in angled brackets; for example, <1.3> means you should have read Section 1.3, Under the Covers, to help you solve this exercise.

Exercise 1.1

Find the word or phrase from the list below that best matches the description in the following questions. Use the numbers to the left of the words in the answer. Each answer should be used only once.

1. virtual worlds	14. operating system
2. desktop computers	15. compiler
3. servers	16. bit
4. low-end servers	17. instruction
5. supercomputers	18. assembly language
6. terabyte	19. machine language
7. petabyte	20. C
8. datacenters	21. assembler
9. embedded computers	22. high-level language
10. multicore processors	23. system software
11. VHDL	24. application software
12. RAM	25. cobol
13. CPU	26. fortran

1.1.1 [2] <1.1> Computer used to run large problems and usually accessed via a network

1.1.2 [2] <1.1> 10^{15} or 2^{50} bytes

1.1.3 [2] <1.1> Computer composed of hundreds to thousands of processors and terabytes of memory

1.1.4 [2] <1.1> Today's science fiction application that probably will be available in near future

1.1.5 [2] <1.1> A kind of memory called random access memory

1.1.6 [2] <1.1> Part of a computer called central processor unit

1.1.7 [2] <1.1> Thousands of processors forming a large cluster

1.1.8 [2] <1.1> A microprocessor containing several processors in the same chip

1.1.9 [2] <1.1> Desktop computer without screen or keyboard usually accessed via a network

1.1.10 [2] <1.1> Currently the largest class of computer that runs one application or one set of related applications

1.1.11 [2] <1.1> Special language used to describe hardware components

package has been opened or removed

- 1.1.12** [2] <1.1> Personal computer delivering good performance to single users at low cost
- 1.1.13** [2] <1.2> Program that translates statements in high-level language to assembly language
- 1.1.14** [2] <1.2> Program that translates symbolic instructions to binary instructions
- 1.1.15** [2] <1.2> High-level language for business data processing
- 1.1.16** [2] <1.2> Binary language that the processor can understand
- 1.1.17** [2] <1.2> Commands that the processors understand
- 1.1.18** [2] <1.2> High-level language for scientific computation
- 1.1.19** [2] <1.2> Symbolic representation of machine instructions
- 1.1.20** [2] <1.2> Interface between user's program and hardware providing a variety of services and supervision functions
- 1.1.21** [2] <1.2> Software/programs developed by the users
- 1.1.22** [2] <1.2> Binary digit (value 0 or 1)
- 1.1.23** [2] <1.2> Software layer between the application software and the hardware that includes the operating system and the compilers
- 1.1.24** [2] <1.2> High-level language used to write application and system software
- 1.1.25** [2] <1.2> Portable language composed of words and algebraic expressions that must be translated into assembly language before run in a computer
- 1.1.26** [2] <1.2> 10^{12} or 2^{40} bytes

Exercise 1.2

- 1.2.1** [10] <1.3> For a color display using 8 bits for each of the primary colors (red, green, blue) per pixel and with a resolution of 1280×800 pixels, what should be the size (in bytes) of the frame buffer to store a frame?
- 1.2.2** [5] <1.3> If a computer has a main memory of 2 GB, how many frames could it store, assuming the memory contains no other information?

1.2.3 [5] <1.3> If a computer connected to a 1 gigabit Ethernet network needs to send a 256 Kbytes file, how long it would take?

1.2.4 [5] <1.3> Assuming that a cache memory is ten times faster than a DRAM memory, that DRAM is 100,000 times faster than magnetic disk, and that flash memory is 1000 times faster than disk, find how long it takes to read a file from a DRAM, a disk, and a flash memory if it takes 2 microseconds from the cache memory?

Exercise 1.3

Consider three different processors P1, P2, and P3 executing the same instruction set with the clock rates and CPIs given in the following table.

Processor	Clock rate	CPI
P1	2 GHz	1.5
P2	1.5 GHz	1.0
P3	3 GHz	2.5

1.3.1 [5] <1.4> Which processor has the highest performance?

1.3.2 [5] <1.4> If the processors each execute a program in 10 seconds, find the number of cycles and the number of instructions.

1.3.3 [10] <1.4> We are trying to reduce the time by 30% but this leads to an increase of 20% in the CPI. What clock rate should we have to get this time reduction?

For problems below, use the information in the following table.

Processor	Clock rate	No. instructions	Time
P1	2 GHz	20×10^9	7 s
P2	1.5 GHz	30×10^9	10 s
P3	3 GHz	90×10^9	9 s

1.3.4 [10] <1.4> Find the IPC (instructions per cycle) for each processor.

1.3.5 [5] <1.4> Find the clock rate for P2 that reduces its execution time to that of P1.

1.3.6 [5] <1.4> Find the number of instructions for P2 that reduces its execution time to that of P3.

Exercise 1.4

Consider two different implementations of the same instruction set architecture. There are four classes of instructions, A, B, C, and D. The clock rate and CPI of each implementation are given in the following table.

	Clock rate	CPI Class A	CPI Class B	CPI Class C	CPI Class D
P1	1.5 GHz	1	2	3	4
P2	2 GHz	2	2	2	2

1.4.1 [10] <1.4> Given a program with 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?

1.4.2 [5] <1.4> What is the global CPI for each implementation?

1.4.3 [5] <1.4> Find the clock cycles required in both cases.

The following table shows the number of instructions for a program.

Arith	Store	Load	Branch	Total
500	50	100	50	700

1.4.4 [5] <1.4> Assuming that arith instructions take 1 cycle, load and store 5 cycles and branch 2 cycles, what is the execution time of the program in a 2 GHz processor?

1.4.5 [5] <1.4> Find the CPI for the program.

1.4.6 [10] <1.4> If the number of load instructions can be reduced by one-half, what is the speed-up and the CPI?

Exercise 1.5

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. The clock rate and CPI of each class is given below.

		Clock rate	CPI Class A	CPI Class B	CPI Class C	CPI Class D	CPI Class E
a.	P1	1.0 GHz	1	2	3	4	3
	P2	1.5 GHz	2	2	2	4	4
b.	P1	1.0 GHz	1	1	2	3	2
	P2	1.5 GHz	1	2	3	4	3

1.5.1 [5] <1.4> 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?

1.5.2 [5] <1.4> If the number of instructions executed in a certain program is divided equally among the classes of instructions except for class A, which occurs twice as often as each of the others. Which computer is faster? How much faster is it?

1.5.3 [5] <1.4> If the number of instructions executed in a certain program is divided equally among the classes of instructions except for class E, which occurs twice as often as each of the others? Which computer is faster? How much faster is it?

The table below shows instruction-type breakdown for different programs. Using this data, you will be exploring the performance tradeoffs with different changes made to a MIPS processor.

		# Instructions				
		Compute	Load	Store	Branch	Total
a.	Program 1	1000	400	100	50	1550
b.	Program 4	1500	300	100	100	1750

1.5.4 [5] <1.4> Assuming that computes take 1 cycle, loads and store instructions take 10 cycles, and branches take 3 cycles, find the execution time of each program on a 3 GHz MIPS processor.

1.5.5 [5] <1.4> Assuming that computes take 1 cycle, loads and store instructions take 2 cycles, and branches take 3 cycles, find the execution time of each program on a 3 GHz MIPS processor.

1.5.6 [5] <1.4> Assuming that computes take 1 cycle, loads and store instructions take 2 cycles, and branches take 3 cycles, what is the speed-up of a program if the number of compute instruction can be reduced by one-half?

Exercise 1.6

Compilers can have a profound impact on the performance of an application on a given processor. This problem will explore the impact compilers have on execution time.

package has been opened or removed.

	Compiler A		Compiler B	
	# Instructions	Execution time	# Instructions	Execution time
a.	1.00E+09	1 s	1.20E+09	1.4 s
b.	1.00E+09	0.8 s	1.20E+09	0.7 s

1.6.1 [5] <1.4> For the same program, two different compilers are used. The table above shows the execution time of the two different compiled programs. Find the average CPI for each program given that the processor has a clock cycle time of 1 nS.

1.6.2 [5] <1.4> Assume the average CPIs found in 1.6.1, but that 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?

1.6.3 [5] <1.4> A new compiler is developed that uses only 600 million instructions and has an average CPI of 1.1. What is the speed-up of using this new compiler versus using Compiler A or B on the original processor of 1.6.1?

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, and P2 has a clock rate of 6 GHz. The average number of cycles for each instruction class for P1 and P2 are listed in the following table.

	Class	CPI on P1	CPI on P2
a.	A	1	2
	B	2	2
	C	3	2
	D	4	4
	E	5	4

	Class	CPI on P1	CPI on P2
b.	A	1	2
	B	1	2
	C	1	2
	D	4	4
	E	5	4

1.6.
a cor
of P

1.6.
divic
class
than

1.6.
the i

Exc
The
of Ir

Pe
P

1.7.
gene
Sect

1.7.
gene

1.7.
with

1.6.4 [5] <1.4> 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?

1.6.5 [5] <1.4> If the number of instructions executed in a certain program is divided equally among the classes of instructions in Problem 2.36.4 except for class A, which occurs twice as often as each of the others, how much faster is P2 than P1?

1.6.6 [5] <1.4> At what frequency does P2 have the same performance as P1 for the instruction mix given in 1.6.5?

Exercise 1.7

The following table shows the increase in clock rate and power of eight generations of Intel processors over 28 years.

Processor	clock rate	Power
80286 (1982)	12.5 MHz	3.3 W
80386 (1985)	16 MHz	4.1 W
80486 (1989)	25 MHz	4.9 W
Pentium (1993)	66 MHz	10.1 W
Pentium Pro (1997)	200 MHz	29.1 W
Pentium 4 Willamette (2001)	2 GHz	75.3 W
Pentium 4 Prescott (2004)	3.6 GHz	103 W
Core 2 Ketsfield (2007)	2.667 GHz	95 W

1.7.1 [5] <1.5> What is the geometric mean of the ratios between consecutive generations for both clock rate and power? (The geometric mean is described in Section 1.7.)

1.7.2 [5] <1.5> What is the largest relative change in clock rate and power between generations?

1.7.3 [5] <1.5> How much larger is the clock rate and power of the last generation with respect to the first generation?

Consider the following values for voltage in each generation.

Processor	Voltage
80286 (1982)	5
80386 (1985)	5
80486 (1989)	5
Pentium (1993)	5
Pentium Pro (1997)	3.3
Pentium 4 Willamette (2001)	1.75
Pentium 4 Prescott (2004)	1.25
Core 2 Ketsfield (2007)	1.1

1.7.4 [5] <1.5> Find the average capacitive loads, assuming a negligible static power consumption.

1.7.5 [5] <1.5> Find the largest relative change in voltage between generations.

1.7.6 [5] <1.5> Find the geometric mean of the voltage ratios in the generations since the Pentium.

Exercise 1.8

Suppose we have developed new versions of a processor with the following characteristics.

Version	Voltage	Clock rate
version 1	5 V	0.5 GHz
version 2	3.3 V	1 GHz

1.8.1 [5] <1.5> By how much has the capacitive load been reduced between versions if the dynamic power has been reduced by 10%?

1.8.2 [5] <1.5> By how much has the dynamic power been reduced if the capacitive load does not change?

1.8.3 [5] <1.5> Assuming that the capacitive load of version 2 is 80% the capacitive load of version 1, find the voltage for version 2 if the dynamic power of version 2 is reduced by 40% from version 1.

Supp
follow

1.8.4

1.8.5

1.8.6
the C

Exe

Altho
leakag
dimer
follow
proce

a.
b.

1.9.1
static

1.9.2
find t

1.9.3
techn

Consi
proce

a.
b.

Supposing that the industry trends show that a new process generation scales as follows:

Capacitance	Voltage	Clock rate	Area
1	$1/2^{-1/4}$	$2^{1/2}$	$2^{-1/2}$

1.8.4 [5] <1.5> By what factor does the dynamic power scales?

1.8.5 [5] <1.5> Find the scaling of the capacitance per unit area.

1.8.6 [5] <1.5> Using data from Exercise 1.7, find the voltage and clock rate of the Core 2 processor for the next process generation.

Exercise 1.9

Although the dynamic power is the primary source of power dissipation in CMOS, leakage current produces a static power dissipation $V \times I_{\text{leak}}$. The smaller the on-chip dimensions, the more significant is the static power. Assume the figures shown in the following table for static and dynamic power dissipation for several generations of processors.

	Technology	Dynamic power (W)	Static power (W)	Voltage (V)
a.	250 nm	49	1	3.3
b.	90 nm	75	45	1.1

1.9.1 [5] <1.5> Find the percentage of the total dissipated power comprised by static power.

1.9.2 [5] <1.5> If the static power depends on the leakage current, $P_{\text{st}} = V \times I_{\text{leak}}$, find the leakage current for each technology.

1.9.3 [5] <1.5> Determine the ratio of static power to dynamic power for each technology.

Consider now the dynamic power dissipation of different versions of a given processor for three different voltages given in the following table.

	1.2 V	1.0 V	0.8 V
a.	80 W	70 W	40 W
b.	65 W	55 W	30 W

1.9.4 [5] <1.5> Determine the static power for each version at 0.8 V, assuming a static to dynamic power ratio of 0.6.

1.9.5 [5] <1.5> Find the leakage current for each version at 0.8 V.

1.9.6 [10] <1.5> Determine the larger of the two leakage currents at 1.0 V and 1.2 V, assuming a static to dynamic power ratio of 1.7.

Exercise 1.10

The table below shows the instruction type breakdown of a given application executed on 1, 2, 4, or 8 processors. Using this data, you will be exploring the speed-up of applications on parallel processors.

	Processors	# Instructions per processor			CPI		
		Arithmetic	Load/Store	Branch	Arithmetic	Load/Store	Branch
a.	1	2560	1280	256	1	4	2
	2	1280	640	128	1	4	2
	4	640	320	64	1	4	2
	8	320	160	32	1	4	2
	Processors	# Instructions per processor			CPI		
		Arithmetic	Load/Store	Branch	Arithmetic	Load/Store	Branch
b.	1	2560	1280	256	1	4	2
	2	1350	800	128	1	6	2
	4	800	600	64	1	9	2
	8	600	500	32	1	13	2

1.10.1 [5] <1.4, 1.6> The table above shows the number of instructions required per processor to complete a program on a multiprocessor with 1, 2, 4, or 8 processors. What is the total number of instructions executed per processor? What is the aggregate number of instructions executed across all processors?

1.10.2 [5] <1.4, 1.6> Given the CPI values on the right of the table above, find the total execution time for this program on 1, 2, 4, and 8 processors. Assume that each processor has a 2 GHz clock frequency.

1.10.3 [10] <1.4, 1.6> If the CPI of arithmetic instructions was doubled, what would the impact be on the execution time of the program on 1, 2, 4, or 8 processors?

The t
proce
Using
proce

a.

b.

1.10.4
time of

1.10.5
can be

where
equatio

with the
the pow
that eac
consum
core is o

1.10.6
and 8 co
Assume

The table below shows the number of instruction per processor core on a multicore processor as well as the average CPI for executing the program on 1, 2, 4, or 8 cores. Using this data, you will be exploring the speed-up of applications on multicore processors.

	Cores per processor	Instructions per core	Average CPI
a.	1	1.00E+10	1.2
	2	5.00E+09	1.3
	4	2.50E+09	1.5
	8	1.25E+09	1.8
	Cores per processor	Instructions per core	Average CPI
b.	1	1.00E+10	1.2
	2	5.00E+09	1.2
	4	2.50E+09	1.2
	8	1.25E+09	1.2

1.10.4 [10] <1.4, 1.6> Assuming a 3 GHz clock frequency, what is the execution time of the program using 1, 2, 4, or 8 cores.

1.10.5 [10] <1.5, 1.6> Assume that the power consumption of a processor core can be described by the following equation

$$\text{Power} = \frac{5.0\text{mA}}{\text{MHz}} \text{Voltage}^2$$

where the operation voltage of the processor is described by the following equation

$$\text{Voltage} = \frac{1}{5} \text{Frequency} + 0.4$$

with the frequency measured in GHz. So, at 5 GHz, the voltage would be 1.4 V. Find the power consumption of the program executing on 1, 2, 4, and 8 cores assuming that each core is operating at a 3 GHz clock frequency. Likewise, find the power consumption of the program executing on 1, 2, 4, or 8 cores assuming that each core is operating at 500 MHz.

1.10.6 [10] <1.5, 1.6> Find the energy required to execute the program for 1, 2, 4, and 8 cores assuming that each core has a clock frequency of 3 GHz and 500 MHz. Assume the power consumption equations from 1.10.5.

Exercise 1.11

The following table shows manufacturing data for various processors.

	Wafer diameter	Dies per wafer	Defects per unit area	cost per wafer
a.	15 cm	90	0.018 defects/cm ²	10
b.	25 cm	140	0.024 defects/cm ²	20

1.11.1 [10] <1.7> Find the yield.

1.11.2 [5] <1.7> Find the cost per die.

1.11.3 [10] <1.7> If the number of dies per wafer is increased by 10% and the defects per area unit increases by 15%, find the die area and yield.

Suppose that, with the evolution of the electronic devices manufacturing technology, the yield varies as shown in the following table.

	T1	T2	T3	T4
yield	0.85	0.89	0.92	0.95

1.11.4 [10] <1.7> Find the defects per area unit for each technology given a die area of 200 mm².

1.11.5 [5] <1.7> Represent graphically the variation of the yield together with the variation of defects per unit area.

Exercise 1.12

The following table shows results for SPEC2006 benchmark programs running on an AMD Barcelona.

	Name	Intr. count × 10 ⁹	Execution time (seconds)	Reference time (seconds)
a.	perl	2118	500	9770
b.	mcf	336	1200	9120

1.12.1 [5] <1.7> Find the CPI if the clock cycle time is 0.333 ns.

1.12.2 [5] <1.7> Find the SPEC ratio.

1.12.3 [5] <1.7> For these two benchmarks, find the geometric mean.

The following table shows data for further benchmarks.

	Name	CPI	Clock rate	SPECratio
a.	sjeng	0.96	4 GHz	14.5
b.	omnetpp	2.94	4 GHz	9.1

1.12.4 [5] <1.7> Find the increase in CPU time if the number of instruction of the benchmark is increased by 10% without affecting the CPI.

1.12.5 [5] <1.7> Find the increase in CPU time if the number of instruction of the benchmark is increased by 10% and the CPI is increased by 5%.

1.12.6 [5] <1.7> Find the change in the SPECratio for the change described in 1.12.5.

Exercise 1.13

Suppose that we are developing a new version of the AMD Barcelona processor with a 4 GHz clock rate. We have added some additional instructions to the instruction set in such a way that the number of instructions has been reduced by 15% from the values shown for each benchmark in Exercise 1.12. The execution times obtained are shown in the following table.

	Name	Execution time (seconds)	Reference time (seconds)	SPECratio
a.	perl	450	9770	21.7
b.	mcf	1150	9120	7.9

1.13.1 [10] <1.8> Find the new CPI.

1.13.2 [10] <1.8> In general, these CPI values are larger than those obtained in previous exercises for the same benchmarks. This is due mainly to the clock rate used in both cases, 3 GHz and 4 GHz. Determine whether the increase in the CPI is similar to that of the clock rate. If they are dissimilar, why?

1.13.3 [5] <1.8> By how much has the CPU time been reduced?

The following table shows data for further benchmarks.

	Name	Execution time (seconds)	CPI	Clock rate
a.	sjeng	820	0.96	3 GHz
b.	omnetpp	580	2.94	3 GHz

1.13.4 [10] <1.8> If the execution time is reduced by an additional 10% without affecting the CPI and with a clock rate of 4 GHz, determine the number of instructions.

1.13.5 [10] <1.8> Determine the clock rate required to give a further 10% reduction in CPU time while maintaining the number of instructions and CPI unchanged.

1.13.6 [10] <1.8> Determine the clock rate if the CPI is reduced by 15% and the CPU time by 20% while the number of instructions is unchanged.

Exercise 1.14

Section 1.8 cites as a pitfall the utilization of a subset of the performance equation as a performance metric. To illustrate this, consider the following data for the execution of given instruction sequence of 10^6 instructions in different processors.

Processor	Clock rate	CPI
P1	4 GHz	1.25
P2	3 GHz	0.75

1.14.1 [5] <1.8> One usual fallacy is to consider the computer with the largest clock rate as having the large performance. Check if this is true for P1 and P2.

1.14.2 [10] <1.8> Another fallacy is to consider that the processor executing the largest number of instruction will need a larger CPU time. Considering that processor P1 is executing a sequence of 10^6 instructions and that the CPI of processors P1 and P2 do not change, determine the number of instructions that P2 can execute in the same time that P1 needs to execute 10^6 instructions.

1.14.3 [10] <1.8> A common fallacy is to use MIPS (millions of instructions per second) to compare the performance of two different processors, and consider that the processor with the largest MIPS has the largest performance. Check if this is true for P1 and P2.

Another common performance figure is MFLOPS (million of floating-point operations per second), defined as

$$\text{MFLOPS} = \text{No. FP operations} / \text{execution time} \times 10^6$$

but this figure has the same problems as MIPS. Consider the programs in the following table, running on a processor with clock rate = 3 GHz.

	Instr. count	L/S Instr.	FP Instr.	Branch Instr.	CPI(L/S)	CPI(FP)	CPI(Branch)
a.	10^6	50%	40%	10%	0.75	1	1.5
b.	3×10^6	40%	40%	20%	1.25	0.70	1.25

1.14.4

1.14.5

1.14.6
MIPS at

Exerc

Another
of a con
but not
the foll

a.	
b.	

1.15.1
operati1.15.2
time is1.15.3
time foThe fol
applica

a.	
b.	

Assum

1.15.4
we war1.15.5
we war

1.14.4 [10] <1.8> Find the MFLOPS figures for the programs.

1.14.5 [10] <1.8> Find the MIPS figures for the programs.

1.14.6 [10] <1.8> Find the performance for the programs and compare with MIPS and MFLOPS.

Exercise 1.15

Another pitfall cited in Section 1.8 is expecting to improve the overall performance of a computer by improving only one aspect of the computer. This might be true, but not always. Consider a computer running programs with CPU times shown in the following table.

	FP instr.	INT instr.	L/S instr.	Branch instr.	Total time
a.	35 s	85 s	50 s	30 s	200 s
b.	50 s	80 s	50 s	30 s	210 s

1.15.1 [5] <1.8> By how much is the total time reduced if the time for FP operations is reduced by 20%?

1.15.2 [5] <1.8> By how much is the time for INT operations reduced if the total time is reduced by 20%?

1.15.3 [5] <1.8> Can the total time can be reduced by 20% by reducing only the time for branch instructions?

The following table shows the instruction type breakdown per processor of a given application executed in different numbers of processors.

	# Processors	FP instr.	INT instr.	L/S instr.	Branch Instr.	CPI (FP)	CPI (INT)	CPI (L/S)	CPI (Branch)
a.	1	560×10^6	2000×10^6	1280×10^6	256×10^6	1	1	4	2
b.	8	80×10^6	240×10^6	160×10^6	32×10^6	1	1	4	2

Assume that each processor has a 2 GHz clock rate.

1.15.4 [10] <1.8> By how much must we improve the CPI of FP instructions if we want the program to run two times faster?

1.15.5 [10] <1.8> By how much must we improve the CPI of L/S instructions if we want the program to run two times faster?

package has been opened or removed.

1.15.6 [5] <1.8> 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%?

Exercise 1.16

Another pitfall, relating to the execution of programs in multiprocessors systems, is expecting improvement in performance by improving only the execution time of part of the routines. The following table shows the execution time of five routines of a program running on different numbers of processors.

	# Processors	Routine A (ms)	Routine B (ms)	Routine C (ms)	Routine D (ms)	Routine E (ms)
a.	2	20	80	10	70	5
b.	16	4	14	2	12	2

1.16.1 [10] <1.8> Find the total execution time and by how much it is reduced if the time of routines A, C, and E is improved by 15%.

1.16.2 [10] <1.8> By how much is the total time reduced if routine B is improved by 10%?

1.16.3 [10] <1.8> By how much is the total time reduced if routine D is improved by 10%?

Execution time in a multiprocessor system can be split into computing time for the routines plus routing time spent sending data from one processor to another. Consider the execution time and routing time given in the following table. In this case, the routing time is an important component of the total time.

# Processors	Routine A (ms)	Routine B (ms)	Routine C (ms)	Routine D (ms)	Routine E (ms)	Routing (ms)
2	20	78	9	65	4	11
4	12	44	4	34	2	13
8	1	23	3	19	3	17
16	4	13	1	10	2	22
32	2	5	1	5	1	23
64	1	3	0.5	1	1	26

1.16.4 [10] <1.8> For each doubling of the number of processors, determine the ratio of new to old computing time and the ratio of new to old routing time.

1.16.5 [5] <1.8> Using the geometric means of the ratios, extrapolate to find the computing time and routing time in a 128-processor system.

1.16.6 [10] <1.8> Find the computing time and routing time for a system with one processor.

§1.1, page 9: Discussion questions: many answers are acceptable.

§1.3, page 25: Disk memory: nonvolatile, long access time (milliseconds), and cost \$0.20–\$2.00/GB. Semiconductor memory: volatile, short access time (nanoseconds), and cost \$20–\$75/GB.

§1.4, page 31: 1. a: both, b: latency, c: neither. 2. 7 seconds.

§1.4, page 38: b.

§1.7, page 50: 1, 3, and 4 are valid reasons. Answer 5 can be generally true because high volume can make the extra investment to reduce die size by, say, 10% a good economic decision, but it doesn't have to be true.

§1.8, page 53: a. Computer A has the higher MIPS rating. b. Computer B is faster.

**Answers to
Check Yourself**

package has been opened or removed.

Improved
L/S and

systems,
time of
routines

routine E
(ms)

5
2

duced if

Improved

Improved

ime for
another.
. In this

outing
(ms)

11
13
17
22
23
26