

Answers to Exercises

Chapter 1 Exercises

For Exercises 1–10, match the following people with their accomplishments.

- A. Leibniz
- B. Pascal
- C. Babbage
- D. Lovelace
- E. Hollerith
- F. Byron
- G. Turing
- H. Jacquard

1. What French mathematician built and sold the first gear-driven mechanical machine that did addition and subtraction?

B

2. Who built the first mechanical machine that did addition, subtraction, multiplication, and division?

A

3. Who designed the first mechanical machine that included memory?

C

4. Who was considered the first programmer?

D

5. Who proposed that a punched card be used for counting the census?

E

6. Who edited Babbage's work?

D

7. Who was Ada Lovelace's father?

F

8. Who would have been mentioned in the book the *Code Breakers*?

G

9. Who developed the concept of punched holes used in weaving cloth?

H

10. Who is associated with IBM?

E

For Exercises 11–23, match the hardware listed to the appropriate generation.

- A. First
- B. Second
- C. Third
- D. Fourth
- E. Fifth

- 11. Circuit boards
C
- 12. Transistor
B
- 13. Magnetic core memory
B
- 14. Card input/output
A
- 15. Parallel computing
D
- 16. Magnetic drum
A
- 17. Magnetic tape drives
A
- 18. Integrated circuits
C
- 19. Personal computer
D
- 20. Vacuum tube
A
- 21. Large scale integration
D
- 22. Magnetic disc
B
- 23. Networking
D

For Exercises 24–38, match the software or software concepts listed to the appropriate generation.

- A. First
 - B. Second
 - C. Third
 - D. Fourth
 - E. Fifth
- 24. Assemblers
 - A
- 25. FORTRAN
 - B
- 26. Operating systems
 - C
- 27. Structured programming
 - D
- 28. Time sharing
 - C
- 29. HTML (for Web)
 - E
- 30. Loaders
 - B
- 31. Spreadsheets
 - D
- 32. Word processors
 - D
- 33. Lisp
 - B
- 34. PC-DOS
 - D
- 35. Loaders/linkers bundled into Operating System
 - C
- 36. Java
 - E
- 37. SPSS
 - C
- 38. C++
 - D

Exercises 39–55 are short answer.

- 39.** What do we mean by the statement that the 1980s and 1990s must be characterized by the changing profile of the user?

The original user was the programmer who had a problem to solve. By the 1970s, application programs were being written such that non-programmers could use them to solve problems. With the advent of the personal computer, many people began using the computer for personal correspondence, personal accounts, and games.

- 40.** Distinguish between computing as a tool and computing as a discipline.

Computing as a tool refers to the use of computing by people to solve problems in their professional or personal life. Computing as a discipline refers to the study of the body of knowledge that makes up computer science and/or computer engineering.

- 41.** Is computing a mathematical discipline, a scientific discipline, or an engineering discipline? Explain.

Computing is neither a mathematical discipline nor a scientific discipline nor an engineering discipline; it has roots in all three disciplines. Boolean algebra, logic, and numerical analysis contribute greatly to the foundations of computing. Simulation and model building from the scientific discipline contribute to the foundations of computing. The techniques from engineering of building large systems contribute to the foundations of computing.

- 42.** Distinguish between systems areas and applications areas in computing as a discipline.

The systems areas of computing as a discipline relate to the understanding and building of computer tools: algorithms and data structures, programming languages, (computer) architecture, operating systems, software methodology and engineering, and human-computer communication. The applications areas in computing relate to the computer's use as a tool: numerical computation, databases and informational retrieval, artificial intelligence and robotics, graphics, organizational informatics, and bioinformatics.

- 43.** Define the word *abstraction* and relate it to the drawing in Figure 1.2.

An abstraction is a mental model that removes complex details. An abstraction lets the viewer see only those details that are relevant to the user's view. An engine is a mental model of a car from the builder or mechanic's view. The cockpit is the mental model for an ordinary driver, who does not have to know the details of the engine in order to drive the car.

- 44.** Name the six subject subareas of computer science, called systems areas, that relate to understanding and building computing tools in general.

algorithms and data structures, programming languages, architecture, operating systems, software methodology and engineering, and human-computer communication.

- 45.** Name the six subject subareas of computer science, called applications areas, that relate to the computer's use as a tool.

numerical and symbolic computation, databases and informational retrieval, artificial intelligence and robotics, graphics, organizational informatics, and bioinformatics.

- 46.** Define the word *protocol* and explain how it is used in computing.

A protocol is a code prescribing strict adherence to correct etiquette and procedure (as in a diplomatic exchange). Computing terminology has borrowed the word to describe the correct etiquette for computers to use when communicating with one another.

- 47.** Distinguish between machine language and assembly language.

Machine language is the language that is built into the electrical circuitry of a computer.

Assembly language is a language made up of mnemonic codes that represent machine-language instructions.

Programs written in assembly language are translated into machine language programs by a computer program called an assembler.

- 48.** Distinguish between assembly language and high-level languages.

Whereas assembly language is a language made up of mnemonic codes that represent machine-language instructions, high-level languages use English-like statements to represent a groups of assembly-language statements or machine-language statements. There is a one-to-one correspondence between statements in an assembly language and the statements they represent in machine language. There is a one-to-many correspondence between high-level statements and the corresponding machine-language statements.

- 49.** FORTRAN and COBOL were two high-level languages defined during the second generation of computer software. Compare and contrast these languages in terms of their history and their purpose.

FORTRAN was written at IBM for use in solving science and engineering problems. It was not formally designed and has grown greatly over the years. COBOL was design by a team for business applications and has been relatively stable.

- 50.** Distinguish between an assembler and a compiler.

An assembler translates assembly-language programs into machine code. A compiler translates programs in a high-level language into either assembly-language programs or machine-language programs.

- 51.** Distinguish between a systems programmer and an applications programmer.

A systems programmer writes programs that are tools to help others write programs. An applications programmer writes programs to solve specific problems.

- 52.** What was the rationale behind the development of operating systems?

The human operator was too slow. Computers were idle while the human prepared the next program to be run. The computer has the speed to organize itself.

- 53.** What constitutes systems software?

Utility programs such as loaders and linkers, operating systems, and language translators are systems software.

- 54.** What do the following pieces of software do?

- a.** Loader

A loader puts a program's instructions into memory where they can be executed.

- b.** Linker

A linker is a program that puts pieces of a large program together so that it can be put into memory where it can be executed.

- c.** Editor

A editor is a word processing program that allows the user to enter and edit text.

- 55.** How was the program SPSS different from the programs that came before it?

SPSS was the first application program written so that the non-programmer user could enter data and specify the processing of the data.

Chapter 2 Exercises

For Exercises 1–5, match the following numbers with their definition.

- A.** Number
- B.** Natural number
- C.** Integer number
- D.** Negative number
- E.** Rational number

1. A unit of an abstract mathematical system subject to the laws of arithmetic.
A
2. A natural number, a negative of a natural number, or zero.
C
3. The number zero and any number obtained by repeatedly adding one to it.
B
4. An integer or the quotient of two integers (division by zero excluded).
E
5. A value less than zero, with a sign opposite to its positive counterpart.
D

For Exercises 6–11, match the solution with the problem.

- A. 10001100
 - B. 10011110
 - C. 1101010
 - D. 1100000
 - E. 1010001
 - F. 1111000
6. $1110011 + 11001$ (binary addition)
A
 7. $1010101 + 10101$ (binary addition)
C
 8. $1111111 + 11111$ (binary addition)
B
 9. $1111111 - 111$ (binary subtraction)
F
 10. $1100111 - 111$ (binary subtraction)
D
 11. $1010110 - 101$ (binary subtraction)
E

For Exercises 12–17, mark the answers true and false as follows:

- A. True
- B. False

- 12.** Binary numbers are important in computing because a binary number can be converted into every other base.
B
- 13.** Binary numbers can be read off in hexadecimal but not in octal.
B
- 14.** Starting from left to right, every grouping of four binary digits can be read as one hexadecimal digit.
B
- 15.** A byte is made up of six binary digits.
B
- 16.** Two hexadecimal digits can be stored in one byte.
A
- 17.** Reading octal digits off as binary produces the same result whether read from right to left or left to right.
A

Exercises 18–45 are problems or short answer questions.

- 18.** Distinguish between a natural number and a negative number.
A natural number is 0 and any number that can be obtained by repeatedly adding 1 to it. A negative number is less than 0, and opposite in sign to a natural number, although we usually do not consider negative 0.
- 19.** Distinguish between a natural number and a rational number.
A rational number is an integer or the quotient of integer numbers. (Division by 0 is excluded.) A natural number is 0 and the positive integers. (See also definition in answer to Exercise 18.)
- 20.** Label the following numbers natural, negative, or rational.
- a.** 1.333333
rational
 - b.** $-1/3$
negative, rational
 - c.** 1066
natural
 - d.** $2/5$
rational
 - e.** 6.2
rational

- f. π (pi)
not any listed
21. If 891 is a number in each of the following bases, how many 1s are there?
- a. base 10
891
 - b. base 8
Can't be a number in base 8
 - c. base 12
1261
 - d. base 13
1470
 - e. base 16
2193
22. Express 891 as a polynomial in each of the bases in Exercise 21.
- base 10: $8 * 10^2 + 9 * 10 + 1$
 - base 8: Can't be shown as a polynomial in base 8.
 - base 12: $8 * 12^2 + 9 * 12 + 1$
 - base 13: $8 * 13^2 + 9 * 13 + 1$
 - base 16: $8 * 16^2 + 9 * 16 + 1$
23. Convert the following numbers from the base shown to base 10.
- a. 111 (base 2)
7
 - b. 777 (base 8)
511
 - c. FEC (base 16)
4076
 - d. 777 (base 16)
1911
 - e. 111 (base 8)
73
24. Explain how base 2 and base 8 are related.
Because 8 is a power of 2, base-8 digits can be read off in binary and 3 base-2 digits can be read off in octal.
25. Explain how base 8 and base 16 are related.
8 and 16 are both powers of two.

26. Expand the table on page 40 to include the numbers from 11 through 16.

binary	octal	decimal
000	0	0
001	1	1
010	2	2
011	3	3
100	4	4
101	5	5
110	6	6
111	7	7
1000	10	8
1001	11	9
1010	12	10
1011	13	11
1100	14	12
1101	15	13
1110	16	14
1111	17	15
10000	20	16

27. Expand the table in Exercise 26 to include hexadecimal numbers.

binary	octal	decimal	hexadecimal
000	0	0	0
001	1	1	1
010	2	2	2
011	3	3	3
100	4	4	4
101	5	5	5
110	6	6	6
111	7	7	7
1000	10	8	8
1001	11	9	9
1010	12	10	A
1011	13	11	B
1100	14	12	C
1101	15	13	D
1110	16	14	E
1111	17	15	F
10000	20	16	20

28. Convert the following binary numbers to octal.

a. 111110110

766

b. 1000001

101

c. 10000010

202

d. 1100010

142

29. Convert the following binary numbers to hexadecimal.

a. 10101001

A9

b. 11100111

E7

c. 01101110

6E

d. 01121111

This is not a binary number

30. Convert the following hexadecimal numbers to octal.

a. A9

251

b. E7

347

c. 6E

156

31. Convert the following octal numbers to hexadecimal.

a. 777

1FF

b. 605

185

c. 443

123

d. 521

151

e. 1

1

32. Convert the following decimal numbers to octal.

- a. 901
1605
- b. 321
501
- c. 1492
2724
- d. 1066
2052
- e. 2001
3721

33. Convert the following decimal numbers to binary.

- a. 45
101101
- b. 69
1000101
- c. 1066
10000101010
- d. 99
1100011
- e. 1
1

34. Convert the following decimal numbers to hexadecimal.

- a. 1066
42A
- b. 1939
793
- c. 1
1
- d. 998
3E6
- e. 43
2B

35. If you were going to represent numbers in base 18, what symbols might you use to represent the decimal numbers 10 through 17 other than letters?

Any special characters would work, as well as characters from another alphabet. Let's use # for 16 and @ for 17.

- 36.** Convert the following decimal numbers to base 18 using the symbols you suggested in Exercise 15.
- a.** 1066
354
 - b.** 99099
#@F9
 - c.** 1
1
- 37.** Perform the following octal additions.
- a.** $770 + 665$
1655
 - b.** $101 + 707$
1010
 - c.** $202 + 667$
1071
- 38.** Perform the following hexadecimal additions.
- a.** $19AB6 + 43$
19AF9
 - b.** $AE9 + F$
AF8
 - c.** $1066 + ABCD$
BC33
- 39.** Perform the following octal subtractions.
- a.** $1066 - 776$
70
 - b.** $1234 - 765$
247
 - c.** $7766 - 5544$
2222
- 40.** Perform the following hexadecimal subtractions.
- a.** $ABC - 111$
9AB
 - b.** $9988 - AB$
98DD
 - c.** $A9F8 - 1492$
9566

- 41.** Why are binary numbers important in computing?
Data and instructions are represented in binary inside the computer.
- 42.** A byte contains how many bits?
8
- 43.** How many bytes are there in a 64-bit machine?
8
- 44.** Why do microprocessors such as pagers have only 8-bit machines?
Pagers are not general purpose computers. The programs in pagers can be represented in 8-bit machines.
- 45.** Why is important to study how to manipulate fixed-sized numbers?
It is important to understand how to manipulate fixed-sized numbers because numbers are represented in a computer in fixed-sized format.

Chapter 3 Exercises

For Exercises 1–20, mark the answers true and false as follows:

- A.** True
B. False

- 1.** Lossless compression means the data can be retrieved without losing any of the original information.
A
- 2.** A computer represents information in an analog form.
B
- 3.** A computer must use the binary number system to represent information.
B
- 4.** A digital signal represents one of two values at any point in time.
A
- 5.** Four bits can be used to represent 32 unique things.
B
- 6.** The signed-magnitude representation of numbers has two representations for zero.
A
- 7.** Overflow occurs when the value that we compute cannot fit into the number of bits we have allocated for the result.
A

8. In the ASCII character set, there is no distinction made between uppercase and lowercase letters.
B
9. The Unicode character set includes all of the characters in the ASCII character set.
T
10. Keyword encoding replaces frequently used words with a single character.
T
11. Run-length encoding is very good at compressing English text.
B
12. Huffman encoding uses variable-length binary strings to represent characters.
A
13. An audio signal is digitized by sampling it at regular intervals.
A
14. A CD stores audio information in a binary format.
A
15. The MP3 audio format discards information that can't be heard by humans.
A
16. An RGB value represents a color using three numeric values.
A
17. Indexed color increases the number of colors that can be used in an image, and thus increases the file size.
B
18. Bitmap, GIF, and JPEG are all examples of raster-graphics formats.
A
19. Vector graphics represent images in terms of lines and geometric shapes.
A
20. A keyframe is used in temporal compression to represent the changes from one frame to another.
B

For Exercises 21–26, choose the correct word from the following list.

- A.** Signed-magnitude representation
- B.** Radix
- C.** Frequency of use
- D.** Sampling
- E.** Analog
- F.** Digital

- 21.** _____ data is a continuous representation of information.
E
- 22.** The representation for numbers you've used since grade school is called _____.
A
- 23.** If the number base is other than base ten, we call the decimal point the _____ point.
B
- 24.** _____ data is a discrete representation of information.
F
- 25.** Huffman codes are created based on the _____ of the character.
C
- 26.** An audio signal is digitized by ____ its value at regular intervals.
D

For Exercises 27–30, choose the correct word from the following list.

- A.** Color depth
- B.** Flash
- C.** Resolution
- D.** Temporal compression
- E.** Analog
- F.** Codec

- 27.** The _____ is the amount of data used to represent a color.
A
- 28.** The _____ of an image is determined by the number of pixels used to represent it.
C
- 29.** The most popular vector graphics format used on the Web today is _____.
B

- 30.** _____ is a video compression technique that looks for differences between consecutive frames of a movie.
- D

Exercises 31–83 are problems or short answer questions.

- 31.** What is data compression and why is it an important topic today?

Data compression refers to reducing the amount of space needed to store a piece of data. Although computer storage is relatively cheap, as the amount of data to store keeps increasing, the cost of storage is a factor. However, the most important reason for compressing data is that we share data more and more. The Web and its underlying networks have limitations on bandwidth that define the maximum number of bits or bytes that can be transmitted from one place to another in a fixed amount of time.

- 32.** What is the difference between lossless and lossy data compression?

A lossless data compression is a technique in which no information is lost. A lossy data compression is a technique in which some information may be lost.

- 33.** Why do computers have difficulty with analog information?

Computers are discrete, finite machines. Analog information is continuous and infinite. Thus, computers cannot represent analog information directly; the analog information must be converted into a digital form.

- 34.** Is a clock with a sweeping second hand an analog or a digital device? Explain.

A sweeping second hand is an analog device. The motion of the hand is continuous.

- 35.** What does it mean to digitize something?

Digitizing is the act of breaking continuous information into discrete pieces so that we can represent each piece separately.

- 36.** What is pulse code modulation?

Pulse code modulation is a signal that jumps sharply between two extremes.

- 37.** How many things can be represented with:

a. four bits

16

b. five bits

32

c. six bits

64

d. seven bits

128

38. Although you have been computing simple arithmetic operations since the second grade, take the following small test to confirm that you thoroughly understand operations on signed integers.

Evaluate the following expressions where W is 17, X is 28, Y is -29, and Z is -13.

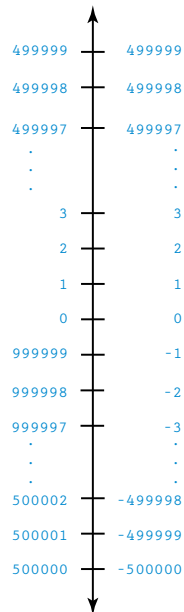
a. $X + Y$	b. $X + W$	c. $Z + W$	d. $Y + Z$
-1	45	4	-42
e. $W - Z$	f. $X - W$	g. $Y - W$	h. $Z - Y$
30	11	-46	16

39. Use the base ten number line to prove the solutions to the following operations, where A is 5 and B is -7.

a. $A + B$ b. $A - B$ c. $B + A$ d. $B - A$



- a. $5 + (-7) = -2$: A is plus 5. To add a -7, you move 7 units to the left from the point of A, landing on -2.
- b. $5 - (-7) = 12$: A is plus 5. To subtract -7, you change the sign of B (7) and move B units to the right from the point of A, landing on 12.
- c. $-7 + 5 = -2$: B is minus 7. To add 5, you move 5 units to the right from the point of B, landing on -2.
- d. $-7 - (5) = -12$: B is minus 7. To subtract 5, you change the sign and move 5 units to the left, landing on -12.
40. Given a fixed-sized number scheme where k in the formula for the ten's complement is 6 (see page 59), answer the following questions.
- a. How many positive integers can be represented?
499,999
- b. How many negative integers can be represented?
500,000
- c. Draw the number line showing the three smallest and largest positive numbers, the three smallest and largest negative numbers, and zero.



- 41.** Use the number line in Exercise 40 to calculate the following expressions, where A is -499999 and B is 3.
- a.** $A + B$ **b.** $A - B$ **c.** $B + A$ **d.** $B - A$
 - a.** -499999 is 500001; move three units to the right, landing on 500004, which is -499996 .
 - b.** -499999 is 500001; move three units to the left, landing outside the range of numbers that can be represented.
 - c.** $+3$ is 3 on the line. -499999 is 500001; move 500001 units to the left, landing on 500004, which is -499996 .
 - d.** $+3$ is 3 on the line. -499999 is 500001; move 500001 units to the right, which is outside the range of numbers that can be represented.
- 42.** Use the formula for the ten's complement to calculate the following numbers in the scheme described on page 59.
- a.** 35768 **b.** -35768 **c.** -444455 **d.** -123456
 - a.** 35768 **b.** 964232 **c.** 555545 **d.** 876544
- 43.** In calculating the ten's complement in Exercise 42, did you have trouble borrowing from so many zeros? Such calculations are error prone. There is a trick that you can use that makes the calculation easier and thus less prone to errors: Subtract from all 9's and then

add 1. A number subtracted from all 9's is called the nine's complement of the number.

- a.** Prove that the nine's complement of a number plus one is equal to the ten's complement of the same number.

Negative(I) = $10^k - I$ in 10's compliment

Negative(I) = $(99..99 - I) + 1$ in 9's compliment.

$(99..99) = (10^k - 1)$

Negative(I) = $(10^k - 1) - I + 1 = 10^k - I$

- b.** Use the nine's complement plus one to calculate the values in Exercise 12 b, c, and d.

b. 964232 **c.** 555545 **d.** 876544

- c.** Which did you find easier to use, the direct calculation of the ten's complement or the nine's complement plus one? Justify your answer.

This is an individual answer.

- 44.** Evaluate the following expressions where A is 11111110 and B is 00000010 using two's complement.

a. $A + B$ **b.** $A - B$ **c.** $B - A$ **d.** $-B$

e. $-(-A)$

a. 00000000 **b.** 11111100 **c.** 00000100 **d.** 11111110

e. 11111110

- 45.** Is the two's complement of a number always a negative number? Explain.

The two's complement is a way of representing fixed-sized numbers in memory. The two's complement of a number X is just $-X$ in two's complement representation. If X is a positive number, the two's complement of X is negative, but if X is negative, the two's complement is a positive number.

- 46.** How is a real value stored in the computer?

Any real value can be described by three properties: the sign (positive or negative), the digits in the value with the radix point assumed to be to the right, and the exponent, which determines how the radix point is shifted relative to the mantissa. A real number is represented in the computer by storing it as an integer along with information showing where the radix point is.

- 47.** Convert the rules for subtraction in a sign-magnitude system to the algorithm format.

Find the first number on the number line
If addition
 Move in the sign direction of second number the specified units
Else
 Move in the opposite sign direction of the second number the specified units

- 48.** Convert the following real numbers to binary (5 binary places).

- a.** 0.50
 0.10000
b. 0.25
 0.01000
c. 0.10
 0.00011

- 49.** Convert the following real numbers to octal (5 octal places).

- a.** 0.50
 0.40000
b. 0.26
 0.20000
c. 0.10
 0.06314

- 50.** Can fractional values be visually converted between octal and binary and back? Explain.

Whole numbers expressed in binary and octal can be converted visually, moving from the radix point to the left. Fractional values can be read by visually moving from the radix point to the right.

- 51.** How many bits would be needed to represent a character set containing 45 characters? Why?

6 bits. 63 distinct things can be represented in 6 bits; 31 distinct things can be represented in 5 bits. Therefore 6 bits must be used.

- 52.** How can the decimal number 175.23 be represented as a sign, mantissa, and exponent?

$$175.23 = 17523 * 10^{-2}$$

- 53.** What is the main difference between the ASCII and Unicode character sets?

The ASCII character set uses 8 bits to represent a character; the Unicode character set uses 16 bits to represent a character. Thus, while ASCII can represent 256 characters, Unicode can represent over 65,000 characters. ASCII is fine for representing English, but the designers of Unicode wanted it to be able to represent all the world's characters. ASCII is a subset of Unicode.

- 54.** Create a keyword encoding table that contains a few simple words. Rewrite a paragraph of your choosing using this encoding scheme. Compute the compression ratio you achieve.

Original text:

Computers are multimedia devices that manipulate data varying in form from numbers to graphics to video. Because a computer can only manipulate binary values, all forms of data must be represented in binary form. Data is classified as being continuous (analog) or discrete (digital).

Decimal values are represented by their binary equivalent, using one of several techniques for representing negative numbers, such as a sign magnitude or two's complement. Real numbers are represented by a triple made up of the sign, the digits in the number, and an exponent that specifies the radix point.

A character set is a list of alphanumeric characters and the codes that represent each one. The most common character set is Unicode (16 bits for each character), which has ASCII as a subset. The 8-bit ASCII set is sufficient for English but not for other (or multiple) languages. There are various ways for compressing text so that it takes less space to store it or less time to transmit it from one machine to another.

Audio information is represented as digitized sound waves. Color is represented by three values that each represent the contribution of red, blue, and green. There are two basic techniques for representing pictures, bitmaps, and vector graphics. Video is broken up into a series of still images, each of which is represented as a picture.

Substitutions:

and:	&	to:	>	the:	~
an:	!	it:	<	is:	=
character:	#	ASCII:	%	that:	\$
represented:	@				

Text with substitutions:

Computers are multimedia devices \$ manipulate data varying in form from numbers > graphics > video. Because a computer can only manipulate binary values, all forms of data must be @ in binary form. Data = classified as being continuous (analog) or discrete (digital).

Decimal values are @ by their binary equivalent, using one of several techniques for representing negative numbers, such as a sign magnitude or one's complement. Real numbers are @ by a triple made up of ~ sign, ~ digits in ~ number, & ! exponent \$ specifies ~ radix point.

A # set = a list of alphanumeric #s & ~ codes \$ represent each one. ~ most common # set = Unicode (16 bits for each #), which has % as a subset. ~ 8-bit % set <\= sufficient for English but not for other (or multiple) languages. There are various ways for compressing text so \$ < takes less space > store < or less time > transmit < from one machine > another.

Audio information = @ as digitized sound waves. Color = @ by three values \$ each represent ~ contribution of red, blue, & green. There are two basic techniques for representing pictures, bitmaps & vector graphics. Video = broken up into a series of still images, each of which = @ as a picture.

Compression ratio: .8864

- 55.** How would the following string of characters be represented using run length encoding? What is the compression ratio?

AAAABBBCCCCCCCCDDDD hi there EEEEEEEEEFF

*A4BBB*C8*D4 hi there *E9FF

Compression ratio is .658

- 56.** What does code *X5*A9 represent using run-length encoding?

XXXXXXXXXXXXXXXX

- 57.** Given the following Huffman encoding table, decipher the bit strings below.

Huffman Code	Character
00	A
11	E
010	T
0110	C
0111	L
1000	S
1011	R
10010	O
10011	I
101000	N
101001	F
101010	H
101011	D

- a. 1101110001011
a. ELATE
b. 0110101010100101011111000
b. CHORES
c. 10100100101000010001000010100110110
c. FANTASTIC
d. 10100010010101000100011101000100011
d. NONSENSE
- 58.** How do humans perceive sound?
We perceive sound when a series of air compressions vibrate a membrane in our ear, which sends signals to our brain.
- 59.** Is a stereo speaker an analog or a digital device? Explain.
A stereo speaker is an analog device because it receives an analog representation of the sound wave from the stereo. The speaker receives the signal and causes a membrane to vibrate, which in turn vibrates the air (creating a sound wave), which in turn vibrates your eardrum.
- 60.** What is an RGB value?
RGB stands for Red, Green, and Blue. The RGB value is actually three numbers that indicate the relative contribution of each of these three colors.

- 61.** What does color depth indicate?

Color depth is the amount of data used to represent a color; that is, the number of bits used to represent each of the colors in the RGB value.

- 62.** How does pixel resolution affect the visual impact of an image?

If enough pixels are used (high resolution) and presented in the proper order side by side, the human eye can be fooled into thinking it's viewing a continuous picture.

- 63.** Explain temporal video compression.

Temporal video compression uses the differences between frames as the basis for compression. A key frame is stored in its entirety, and then changes to that key frame are stored.

- 64.** Describe a situation in which spatial video compression would be effective.

Spatial video compression removes redundant information within a frame. This type of compression is good for landscapes in which there are large blocks of color that are the same, such as a blue sky.

- 65.** Define sampling as it relates to digitizing sound waves.

To digitize the signal, we periodically measure the voltage of the signal and record the appropriate numeric value. Instead of a continuous signal, we have a series of numbers representing distinct voltage values. Thus, we have a sample of the original continuous signal.

- 66.** Which produces better sound quality, higher sampling rates or lower sampling rates?

Higher sampling rates produce better sound quality.

- 67.** What is the sampling rate per second that is enough to create reasonable sound reproduction?

40,000

- 68.** Do vinyl record albums and compact discs record sound the same way?

No. Vinyl record albums use an analog representation, but compact discs store audio information digitally.

- 69.** What does an RGB value of (130, 0, 255) mean?

There is a medium contribution of red, no contribution of green, and a full contribution of blue.

- 70.** What color does an RGB value of (255, 255, 255) represent?

White

- 71.** What is resolution?
Resolution refers to the number of pixels used to represent a picture.
- 72.** The GIF format uses what technique?
GIF uses indexed color.
- 73.** What are GIF files best for?
GIF files are best for graphics and images with few colors.
- 74.** How are the various video codecs alike?
Most video codecs are block-oriented; each frame of a video is divided into rectangular blocks.
- 75.** How are they different?
The codecs differ in how the blocks used are encoded.
- 76.** Name two types of video compression.
Temporal compression and spatial compression
- 77.** What do we call the perception of the various frequencies of light that reach the retinas of our eyes?
Color
- 78.** What is the best format for photographic color images?
JPEG
- 79.** What are the techniques called that shrink the sizes of movies?
Video codecs
- 80.** What is the technique in which an application supports only a certain number of specific colors, creating a palette from which to choose?
Indexed color
- 81.** What is the format that describes an image in terms of lines and geometric shapes?
Vector graphics
- 82.** What format stores information on a pixel-by-pixel basis?
Raster-graphics format
- 83.** What is the difference between HiColor and TrueColor?
HiColor uses 16 bits; TrueColor uses 24 bits.

Chapter 4 Exercises

For Exercises 1–17, mark the answers true and false as follows:

A. True

B. False

- 1.** Logic diagrams and truth tables are equally powerful in expressing the processing of gates and circuits.

A

- 2.** Boolean expressions are more powerful than logic diagrams in expressing the processing of gates and circuits.

B

- 3.** A NOT gate accepts two inputs.

B

- 4.** The output value of an AND gate when both inputs are 1 is 1.

A

- 5.** The AND and OR gates produce opposite results for the same input.

B

- 6.** The output value of an OR gate when both inputs are 1 is 1.

A

- 7.** The output of an OR gate when one input is 0 and one input is 1 is 0.

B

- 8.** The output value of an XOR gate is 0 unless both inputs are 1.

B

- 9.** The NOR gate produces the opposite results of the XOR gate.

B

- 10.** A gate can be designed to accept more than two inputs.

A

- 11.** A transistor is made of semiconductor material.

A

- 12.** Inverting the output of an AND gate is equivalent to inverting the individual signals first, then passing them through an OR gate.

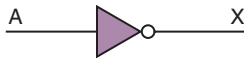
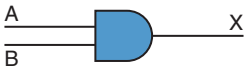
A (DeMorgan's law)

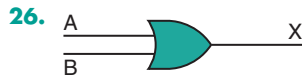
- 13.** The sum of two binary digits (ignoring the carry) is expressed by an AND gate.

B

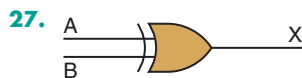
- 14.** A full adder takes the carry-in value into account.
A
- 15.** A multiplexer adds all of the bits on its input lines to produce its output.
B
- 16.** Integrated circuits are classified by the number of gates contained in them.
A
- 17.** A CPU is an integrated circuit.
A

For Exercises 18–29, match the gate with the description of the operation.

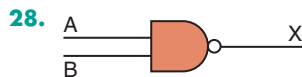
- A.** AND
B. NAND
C. XOR
D. OR
E. NOR
F. NOT
- 18.** Inverts its input.
F
- 19.** Produces a 1 only if all its inputs are 1 and a 0 otherwise.
A
- 20.** Produces a 0 only if all its inputs are 0 and a 1 otherwise.
D
- 21.** Produces a 0 only if all its inputs are the same and a 1 otherwise.
C
- 22.** Produces a 0 if all its inputs are 1 and a 1 otherwise.
B
- 23.** Produces a 1 if all its inputs are 0 and a 0 otherwise.
E
- 24.** 
F
- 25.** 
A



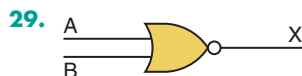
D



C



B



E

Exercises 30–73 are short answer or design questions.

- 30.** How is voltage level used to distinguish between binary digits?

A voltage level in the range of 0 to 2 volts is interpreted as a binary 0.
A voltage level in the range of 2+ to 5 volts is interpreted as a binary 1.

- 31.** Distinguish between a gate and a circuit.

A gate accepts one or more input signals and produces an output signal. Each type of gate performs one logical function. A circuit is a combination of gates designed to accomplish a more complex logical function.

- 32.** What are the three notational methods for describing the behavior of gates and circuits?

Boolean expressions, logic diagrams, and truth tables

- 33.** Characterize the notations asked for in Exercise 32.

Boolean expressions use the operations of Boolean algebra to describe the behavior of gates and circuits.

Logic diagrams use a graphical representation to describe the behavior of gates and circuits.

Truth tables define the behavior of gates and circuits by showing all possible input and output combinations of the gates and circuits.

- 34.** How many input signals can a gate receive and how many output signals can a gate produce?

A gate can accept one or more input signals, but can produce only a single output value.

- 35.** Name six types of gates.

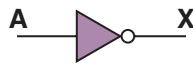
NOT, AND, OR, XOR, NAND, NOR

- 36.** Give the three representations of a NOT gate and say in words what NOT means.

A is the input signal and X is the output signal.

Boolean expression: $X = A'$

Logic Diagram Symbol



Truth Table

A	X
0	1
1	0

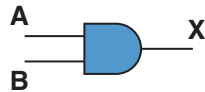
NOT takes a binary input value and inverts it.

- 37.** Give the three representations of an AND gate and say in words what AND means.

A and B are the input signals and X is the output signal.

Boolean expression: $A \cdot B$ (A AND B)

Logic Diagram Symbol



Truth Table

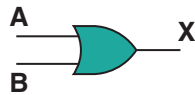
A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

If both input values are 1, AND returns 1; otherwise AND returns 0.

- 38.** Give the three representations of an OR gate and say in words what OR means.

A and B are the input signals and X is the output signal.

Boolean expression: $A + B$ (A OR B)

Logic Diagram Symbol**Truth Table**

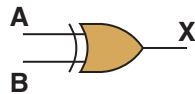
A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

If both input values are 0, OR returns 0; otherwise OR returns 1.

- 39.** Give the three representations of an XOR gate and say in words what XOR means.

A and B are the input signals and X is the output signal.

Boolean expression: $A \oplus B$ (A XOR B)

Logic Diagram Symbol**Truth Table**

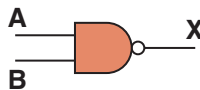
A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

If both inputs are the same value, XOR returns a 0; otherwise XOR returns 1.

- 40.** Give the three representations of a NAND gate and say in words what NAND means.

A and B are the input signals and X is the output signal.

Boolean expression: $(A \cdot B)'$ (NOT (A AND B))

Logic Diagram Symbol**Truth Table**

A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

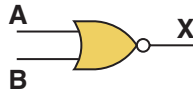
If the inputs are different or both 0, NAND returns 1; if both are 1, it returns 0.

- 41.** Give the three representations of a NOR gate and say in words what NOR means.

A and B are the input signals and X is the output signal.

Boolean expression: $(A + B)'$ (NOT (A AND B))

Logic Diagram Symbol



Truth Table

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

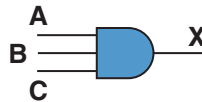
If the inputs are both 0, NOR returns 1; otherwise NOR returns 0.

- 42.** Why are there no logic diagram symbols for the NAND and NOR gates?

Because NAND means not AND and NOR means not OR, there are no symbols for NAND and NOR. The AND and OR symbols are used with the inversion bubble.

- 43.** Draw and label the symbol for a three input AND gate, then show its behavior with a truth table.

Logic Diagram Symbol

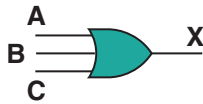


Truth Table

A	B	C	X
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

- 44.** Draw and label the symbol for a three-input OR gate, then show its behavior with a truth table.

Logic Diagram Symbol



Truth Table

A	B	C	X
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

- 45.** What is used in a gate to establish how the input values map to the output value?
A transistor
- 46.** How does a transistor behave?
Depending on the voltage of an input signal, a transistor either acts as a wire that conducts electricity or as a resistor that blocks the flow of electricity.
- 47.** Of what is a transistor made?
Transistors are made of semiconductor material, which is neither a good conductor of electricity nor a particularly good insulator. Transistors are usually made from silicon.
- 48.** What happens when an electric signal is grounded?
If an electric signal is grounded, the signal flows through an alternative route to the ground where it can do no harm. When a signal is grounded, it is pulled down to 0 volts.
- 49.** What are the three terminals in a transistor and how do they operate?
The *source* is an electric signal. The *base* value regulates a gate that determines whether the connection between the source and the ground (*emitter*) is made. An output line is usually connected to the source. If the base value is high, the source is grounded and the output is low (representing 0). If the base value is low, the gate is closed, the source is not grounded, and the output is high (representing 1).
- 50.** How many transistors does it take for each of these gates?
- NOT
1
 - AND
2

c. NOR

2

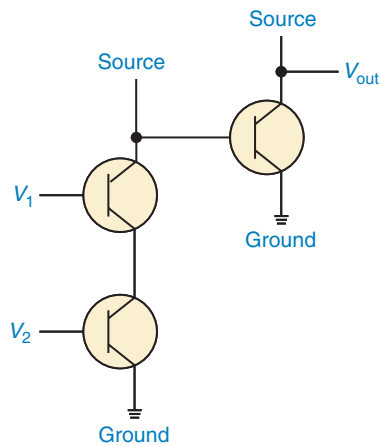
d. OR

2

e. XOR

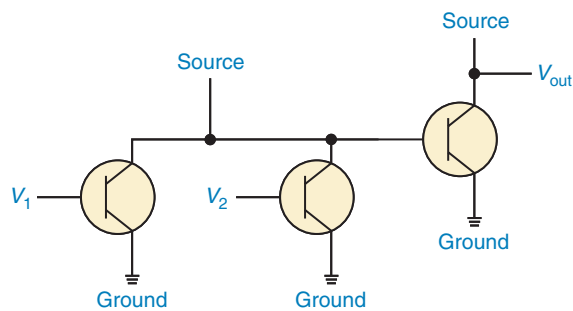
2

51. Draw a transistor diagram for an AND gate. Explain the processing.



The NAND gate is the inverse of the AND gate, and the inverse of the inverse is the original. Thus, the output from the NAND gate is input to a NOT gate, giving us the AND.

52. Draw a transistor diagram for an OR gate. Explain the processing.



The NOR gate is the inverse of the OR gate, and the inverse of the inverse is the original. Thus, the output from the NOR gate is input to a NOT gate, giving us the OR.

- 53.** How can gates be combined into circuits?

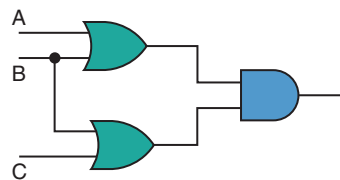
Gates are combined into circuits by using the output of one gate as the input for another. Also, the same input value can be used as input to two different gates.

- 54.** What are the two general categories of circuits and how do they differ?

Combinational circuits are circuits in which the input values explicitly determine the output. *Sequential circuits* are circuits in which the output is a function of input values and the current state of the circuit.

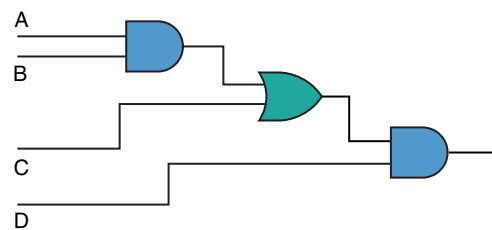
- 55.** Draw a circuit diagram corresponding to the following Boolean expression:

$$(A + B)(B + C)$$



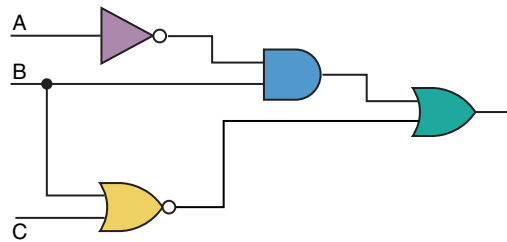
- 56.** Draw a circuit diagram corresponding to the following Boolean expression:

$$(AB + C)D$$



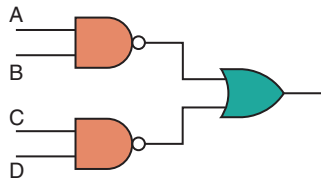
- 57.** Draw a circuit diagram corresponding to the following Boolean expression:

$$A'B + (B+C)'$$

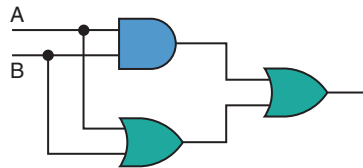


- 58.** Draw a circuit diagram corresponding to the following Boolean expression:

$$(AB)' + (CD)'$$

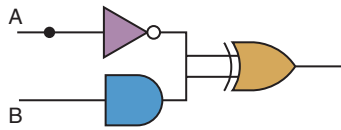


- 59.** Show the behavior of the following circuit with a truth table:



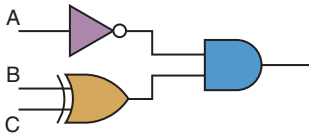
A	B	AB	A+B	AB + (A + B)
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	1

60. Show the behavior of the following circuit with a truth table:



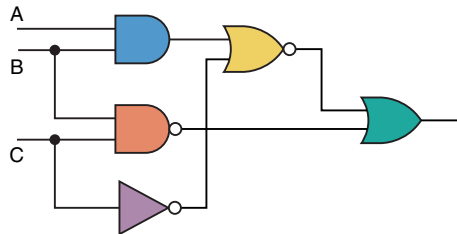
A	B	A'	AB	A'(B ⊕ C)
0	0	1	0	1
0	1	1	0	1
1	0	0	0	0
1	1	0	1	1

61. Show the behavior of the following circuit with a truth table:



A	B	C	A'	B ⊕ C	A'(B ⊕ C)
0	0	0	1	0	0
0	0	1	1	1	1
0	1	0	1	1	1
0	1	1	1	0	0
1	0	0	0	0	0
1	0	1	0	1	0
1	1	0	0	1	0
1	1	1	0	0	0

62. Show the behavior of the following circuit with a truth table:



A	B	C	AB	(BC)'	C'	(AB + C)'	(BC)' + (AB + C)'
0	0	0	0	1	1	0	1
0	0	1	0	1	0	1	1
0	1	0	0	1	1	0	1
0	1	1	0	0	0	1	1
1	0	0	0	1	1	0	1
1	0	1	0	1	0	1	1
1	1	0	1	1	1	0	1
1	1	1	1	0	0	0	0

63. What is circuit equivalence?

Circuit equivalence is when two circuits produce the same output from the same input value combination.

64. Name six properties of Boolean algebra and explain what each means.

Commutative: The commutative property says that binary operations AND and OR may be applied left to right or right to left. (A AND B is the same as B AND A; A OR B is the same as B OR A.)

Associative: The associative property says that given three Boolean variables, they may be ANDed or ORed right to left or left to right. ((A AND B) AND C is the same as A AND (B AND C); (A OR B) OR C is the same as A OR (B OR C).)

Distributive: The distributive property says that given three Boolean variables, the first AND the result of the second OR the third is the same as the first AND the second OR the first AND the third. (A AND (B OR C) = (A AND B) OR (A AND C). Also, the first OR the result of second AND the third is the same as the first OR the second AND the result of the first OR the third. (A OR (B AND C) = (A OR B) AND (A OR C).)

Identity: The identity property says that any value A AND the OR identity always returns A and that any value A OR the AND identity always returns A. (A AND 1 = A; A OR 0 = A.)

Complement: The complement property says that any value AND the complement of that value equals the OR identity and that any value OR the complement of that value equals the OR identity. ($A \text{ AND } (A') = 0$; $A \text{ OR } (A') = 1$.)

DeMorgan's Law: DeMorgan's Law says that the complement of A AND B is the same as the complement of A OR the complement of B, and the complement of A OR B is the same as the complement of B AND the complement of A. ($(A \text{ AND } B)' = A' \text{ OR } B'$; $(A \text{ OR } B)' = A' \text{ AND } B'$.)

- 65.** Differentiate between a half adder and a full adder.

A half adder is a circuit that computes the sum of two bits and produces the appropriate carry bit. A full adder is a circuit that computes the sum of two bits, taking into account the carry bit.

- 66.** What is the Boolean expression for a full adder?

C is the carry in.

Sum is $(A \oplus B) \oplus C$

Carry out is $(A \text{ AND } B) \text{ OR } ((A \oplus B) \text{ AND } C)$

- 67.** What is a multiplexer?

A multiplexer is a circuit that uses input control signals to determine which of several data input lines is to be routed to the output.

- 68. a.** Circuits used for memory are what type of circuits?

Memory circuits are sequential circuits because they are dependent on the existing state of the circuit as well as input to the circuit.

- b.** How many digits does an S-R latch store?

One binary digit

- c.** The design for an S-R latch shown in Figure 4.12 guarantees what about the outputs X and Y?

The values of X and Y are always complements.

- 69.** What is an integrated circuit or chip?

An integrated circuit or chip is a piece of silicon into which many gates have been embedded.

- 70.** Define the abbreviations SSI, MSI, LSI, and VLSI.

Each of these abbreviations refers to the number of gates contained in an integrated circuit.

SSI (Small scale integration): contains 1 to 10 gates.

MSI (Medium scale integration): contains 10 to 100 gates.

- 73.** How can the XOR operation be expressed using other operators?
(A OR B) AND (NOT (A AND B))

Chapter 5 Exercises

For exercises 1–16, match the power of ten to its name or use.

- A.** 10^{-12}
- B.** 10^{-9}
- C.** 10^{-6}
- D.** 10^{-3}
- E.** 10^3
- F.** 10^6
- G.** 10^9
- H.** 10^{12}
- I.** 10^{15}

1. Nano

B

2. Pico

A

3. Micro

C

4. Milli

D

5. Tera

H

6. Giga

G

7. Kilo

E

8. Mega

F

9. Often used to describe processor speed.

G

10. Often used to describe size of memory.

F

11. Used in relation to Internet speeds.

E

12. Latin for thousand.

D

13. Spanish for little.

A

14. Peta

I

15. Roughly equivalent to 2^{10} .

E

16. Roughly equivalent to 2^{50} .

I

For exercises 17–23, match the acronym with its most accurate definition.

A. CD-ROM

B. CD-DA

C. CD-WORM

D. DVD

E. CD-RW

F. CD

17. Generic compact disk that is recorded during manufacturing.

F

18. Data is stored in the sectors reserved for timing information in another variant.

A

19. Can be read many times, but written after its manufacture only once.

C

20. Can be both read from and written to any number of times.

E

21. Format used in audio recordings.

B

22. There is one tract that spirals from the inside out.

F

23. A newer technology that can store large amounts of multimedia data.

D

Exercises 24–64 are problems or short answer exercises.

24. Define the following terms

a. Pentium IV processor

The Pentium IV is a popular central processing unit made by Intel.

- b.** hertz
A hertz is a unit of frequency equal to one cycle per second.
 - c.** random access memory
Random access memory is memory in which each word has an address by which the word can be directly accessed.
- 25.** What is the word length in the Pentium IV processor?
The word length of the Pentium IV processor is 32 bits or 4 bytes.
- 26.** What does it mean to say that a processor is 1.4 GHz?
The speed of the processor is 1,400,000,00 cycles per second.
- 27.** What does it mean to say that memory is 133MHz?
Saying a memory is 133MHz means that the memory can be accessed at 133,000,000 cycles per second.
- 28.** How many bytes of memory are there in the following machines?
 - a.** 128MB machine
 $128 * 2^{20}$
 - b.** 256MB machine
 $256 * 2^{20}$
- 29.** Define RPM and discuss what it means in terms of speed of access to a disk.
RPM stands for revolutions per minute. This is a measure of how fast a disk revolves. Data can only be accessed when the reading head is over the data. Therefore the RPM indicates how fast each piece of data can be accessed.
- 30.** What is the stored-program concept and why is it important?
The stored program concept means that data and instructions are both logically the same and can both be stored in memory. The von Neumann architecture is built around this principle. It is important because the human does not have to execute instruction from without the machine. Instructions can be stored in memory and executed in sequence referencing the data values it needs to operate on.
- 31.** What does “units that process information are separate from the units that store information” mean in terms of a computer architecture?
This expression means that memory is separate from the central processing unit.
- 32.** Name the components of a von Neumann machine.
memory, arithmetic/logic unit, input/output units, the control unit

- 33.** What is the addressability of an 8-bit machine?

8

- 34.** What is the function of the ALU?

The ALU performs basic arithmetic operations (addition, subtraction, multiplication, and division) and logical operations (AND, OR, NOT).

- 35.** Which component in the von Neumann architecture acts as the stage manager. Explain.

The computer component that acts as the state manager is the control unit. It controls the actions of the other components in order to execute instructions in sequence.

- 36.** Punched cards and paper tape were early input/output mediums. Discuss their advantages and disadvantages.

Punched cards and paper tape used for input were prepared on separate machines and then read into the computer. Input from cards and paper tape is slow, but they provided a permanent record of the input. When used for output, cards and paper tape had to be transferred to another device to get a human-readable copy of the information; however, the output could be stored permanently on cards and paper tape.

- 37.** What is an instruction register, and what is its function?

The instruction register is a special register in the control unit. It holds the instruction being executed.

- 38.** What is a program counter, and what is its function?

The program counter is a special register in the control unit. It holds the address of the next instruction to be executed.

- 39.** List the steps in the fetch-execute cycle.

Fetch the next instruction from the address in the program counter.

Decode the instruction.

Execute the instruction.

- 40.** Explain what is meant by “fetch an instruction.”

The control unit goes to the address named in the program counter, makes a copy of the contents of that address, puts the copy into the instruction register, and increments the program counter.

- 41.** Explain what is meant by “decode an instruction.”

The control unit determines what the instruction is and accesses any memory locations that contain operands for the instruction.

- 42.** Explain what is meant by “execute an instruction.”
Signals are sent to the arithmetic/logic unit to carry out the processing.
- 43.** Compare and contrast RAM and ROM memory.
RAM is an acronym for Random Access Memory; ROM is an acronym for Read Only Memory. Both RAM and ROM are random access; that is, each cell in memory is directly accessible. The cells in RAM can be both read from and written to. The cells in ROM can only be read from. The bit pattern in ROM is determined at the time of manufacture or burned when the computer is assembled. Once a ROM has been burned (written), it cannot be changed. Another major difference is that RAM is volatile and ROM is not. This means that RAM does not maintain its bit patterns when the power is turned off, but ROM does.
- 44.** What is a secondary storage device, and why are such devices important?
Because RAM is volatile and ROM cannot be changed, there must be places to store data and programs outside of the computer’s main memory. Such places are called secondary storage devices.
- 45.** Discuss the pros and cons of using magnetic tape as a storage medium.
Magnetic tape is a cheap medium for storing great quantities of data. However, data items cannot be directly addressed on tape. To reach a data object, all information recorded before the one you want must be read and discarded.
- 46.** Draw one surface of a disk showing the tracks and sectors.
Your picture should have concentric circles, getting smaller as they move towards the middle. There should be a series of diameters that cut the circles into sectors.
- 47.** Define what is meant by a block of data.
A block of data is the data stored in a sector of a disk.
- 48.** What is a cylinder?
A cylinder is a set of concentric tracks; that is, tracks that line up under one another.
- 49.** Define the steps that a hard disk drive goes through to transfer a block of data from the disk to memory.
To access a block of data, the read/write head moves to the proper track, waits until the proper sector rotates beneath it, and then accesses the data.

- 50.** Distinguish between a compact disk and a magnetic disk.

Data is stored optically on a compact disk and magnetically on a magnetic disk. Rather than having a series of concentric tracks, a compact disk has one track that spirals from inside out. Both disks have the track broken into sectors. The density at which the data is packed varies in a magnetic disk, but does not in a compact disk. The rotation speed of a magnetic disk reader is constant, but the rotation speed of a compact disk varies depending on the position of the laser beam.

- 51.** Describe a parallel architecture that uses synchronous processing.

There are multiple processors applying the same program to multiple data sets.

- 52.** Describe a parallel architecture that uses pipeline processing.

Multiple processors are arranged in tandem. Each processor contributes one part of the overall processing.

- 53.** How does a shared memory parallel configuration work?

Multiple processors do different processing with different data but communicate through the use of shared global memory.

- 54.** How many different memory locations can a 16-bit processor access?

2^{16} different memory locations

- 55.** In discussing the computer ad, we used the expression “Faster is better” three times. Explain what it means in each case.

The faster the speed of the processor, the more powerful the machine is.

The faster data can be accessed from memory, the faster the machine operates.

The faster data can be accessed from a hard drive, the faster the machine operates.

- 56.** In discussing the computer ad, we used the expression “Smaller is better” in relation to the monitor. Explain.

The dot pitch refers to the distance between dots on the screen. The smaller the distance, the clearer the picture on the screen.

- 57.** In discussing the computer ad, we used the expression “Bigger is better” in relation to the compact disc. Explain.

The bigger the external storage device, the more information that can be stored.

58. Keep a diary of how many times during one week the terms hardware and software appear in television commercials.

No answer expected.

59. Take a current ad for a desktop computer and compare that ad with the one shown at the beginning of this chapter.

Answer varies with each student.

60. What is the common name for the disk that is a secondary storage device?

Although there are a variety of disks that are secondary storage devices, the hard disk drive is the one that comes with the machine.

61. To what does the expression *dot pitch* refer?

The distance between the dots on the screen.

62. What is a modem?

A modem is a type of device that allows you to connect to the Internet.

63. Which are faster, download speeds or upload speeds?

Download

64. What is the difference between 1K of memory and 1K transfer rate.

1K of memory is 1024 bytes of memory; 1K transfer rate is 1000/bits per some time measure.

Chapter 6 Exercises

For Exercises 1–10, match the activity with the phase of the object-oriented methodology.

- A. Brainstorming
 - B. Filtering
 - C. Scenarios
 - D. Responsibility algorithms
1. Reviewing list of possible classes, looking for duplicates or missing classes.
B
 2. Asking “what if” questions.
C
 3. Assigning responsibilities to classes.
C

4. Generating first approximation to the list of classes in a problem.
A
5. Assigning collaborators to a responsibility.
C
6. Developing algorithms for the responsibilities listed on a CRC card.
D
7. Output from this phase is a fully developed CRC card for all classes.
C
8. Output from this phase is the design ready to be translated into a program.
D
9. During this phase, inheritance relationships are explored.
B
10. Phase in which functional programming techniques are appropriate.
D

For Exercises 11–16, match the word with its definition.

- A. Information hiding
 - B. Abstraction
 - C. Data abstraction
 - D. Procedural abstraction
 - E. Control abstraction
 - F. Encapsulation
11. What is the bundling of data and actions so that the logical properties of data and actions are separated from their implementation?
F
 12. What is the practice of hiding the details of a module with the goal of controlling access to the details of the module?
A
 13. What is a model of a complex system that includes only the details essential to the viewer?
B
 14. What is the separation of the logical view of an action from its implementation?
D
 15. What is the separation of the logical view of a control structure from its implementation?
E

- 16.** What is the separation of the logical view data from its implementation?

C

Exercises 17–57 are short answer or designs.

- 17.** List the four steps in Polya's How To Solve It List.

Understanding the problem

Devising a plan

Carrying out the plan

Looking back

- 18.** Describe the four steps listed in Exercise 1 in your own words.

Each student's answer is unique.

- 19.** List the problem-solving strategies discussed in this chapter.

Ask questions

Look for familiar things

Divide and conquer

- 20.** Apply the problem-solving strategies to the following situations.

Solutions are not unique.

- a.** Buying a toy for your four-year-old cousin.

Ask questions:

What do four-year olds like?

Is he or she into sports?

What stores sell toys?

Where is a particular store located?

What toys does the cousin already have?

Look for things that are familiar:

I liked Lincoln Logs; would my cousin?

I liked my red wagon; would my cousin?

My cousin is like his (or her) mother; what did she play with as a child?

Divide and conquer:

Go to store.

Go to toy aisle.

Find girl's (or boy's) toys.

Choose one.

- b.** Organizing an awards banquet for your soccer team.

Ask questions:

Where will it be?

When will it be?

How many people will be there?

How many trophies will be awarded?

Look for things that are familiar:

I organized one last year.

I organized a fundraiser.

I was a scout leader.

I play soccer.

Divide and conquer:

Have Jane decide on day and time.

Have Jim choose menu.

Have Mary buy trophies.

Have Jeremy call people.

- c.** Buying a dress or suit for an awards banquet at which you are being honored.

Ask Questions:

What time of day is the banquet?

Where is the banquet being held?

What will others be wearing?

What is my best color?

Look for things that are familiar:

Last year the award winner wore a blue dress (suit).

Last year I wore a green suit.

I wore a suit when I was honored last year.

Divide and conquer:

Choose the store.

Go to the store.

Choose possibles from racks.

Choose one.

- 21.** Examine the solutions in Exercise 4 and determine three things they have in common.

Each solution includes data objects: toy, food, dress, suit.

Each solution involves choices or decisions.

Each solution involves a container for objects: toy store, restaurant, clothing store.

- 22.** What is an algorithm?

An algorithm is a set of instructions for solving a problem in a finite amount of time using a finite amount of data.

- 23.** Write an algorithm for the following tasks.

Solutions are not unique.

- a.** Making a peanut butter and jelly sandwich.

```
Get bread
Get peanut butter
Get jelly
Get knife
Spread peanut butter on one slice of bread
Spread jelly on one slice of bread
Combine slices of bread, peanut butter facing jelly
```

- b.** Getting up in the morning.

```
Alarm goes off
Hit sleep button
Alarm goes off
Hit sleep button
Alarm goes off
Turn off alarm
Move dog
Throw back covers
Put feet over side of the bed
Stand up
```

c. Doing your homework

Turn off TV
Turn off CD
Get backpack
Sit at desk
Open backpack
Pet cat
Open book
Open assignment
While (more to do)
 Solve problem
 Pet cat

d. Driving home in the afternoon

Find car
Open car door
Get into car
Fasten seat belt
Start engine
Turn on radio
While (not yet home)
 Keep going
Turn off engine
Open car door
Get out of car
Close car door

- 24.** List the three phases of the computer problem-solving model.

Algorithm development phase

Implementation phase

Maintenance phase

- 25.** How does the computer problem-solving model differ from Polya's?

In Polya's list, the human executes the plan and evaluates the results.

In a computer solution, a program is written that expresses the plan in

a language that the computer can execute. The human then takes the computer output and evaluates the results.

- 26.** Describe the steps in the algorithm development phase.

The algorithm development phase includes analysis (understanding the problem), proposed solution (logical sequence of solution steps), and testing (following algorithm).

- 27.** Describe the steps in the implementation phase.

The implementation phase includes coding (translating the algorithm into a computer language) and testing (compiling and running the program).

- 28.** Describe the steps in the maintenance phase.

The maintenance phase involves using the program and modifying the program to add functionality or correct errors.

- 29.** Look up a recipe for chocolate brownies in a cookbook and answer the following questions.

- a.** Is the recipe an algorithm? Justify your answer.

(One author's solution.)

Yes, the recipe is an algorithm. If the steps are followed exactly, brownies are produced.

- b.** Organize the recipe as an algorithm, using pseudo-code.

Preheat oven to 375°
Put 2 oz unsweetened chocolate in double boiler
Add 1/2 cup butter to chocolate in double boiler
Put double boiler over moderate flame
Melt contents of double boiler
Remove double boiler from flame
Get a cup of sugar
Put 2 eggs in bowl
While (more sugar)
 Beat eggs
 add sugar gradually
Put contents of cooled double boiler in bowl
Mix contents of bowl
Sift 1/2 cup flour and dash of salt
Stir flour mixture into bowl
Add 1 teaspoon vanilla to bowl
Add 1/2 cup chopped nuts to bowl
Mix contents of bowl
Grease 9-inch square pan
Pour contents of bowl into pan
Set minutes to 20
Put pan in oven
While (minutes not 0)
 Set minutes to minutes - 1
Remove pan from oven
Cut into 1-1/2" squares
Eat

- c. List the words that have meaning in computing.

While is the only computing word. It means repetition.

- d. List the words that have meaning in cooking.

Words with meaning in cooking include preheat, add, double boiler, melt, moderate flame, beat, gradually, mix, sift, dash, chopped, and grease.

- e. Make the brownies and take them to your professor.

- 30.** We said that following a recipe is easier than developing one. Go to the supermarket and buy a vegetable that you have not cooked (or eaten) before. Take it home and develop a recipe. Write up your recipe and your critique of the process. (If it is good, send it to the authors.)

This is an activity. No answer expected.

- 31.** Describe the top-down design process.

The top-down design process is characterized by successive layers of refinement. The top-level tasks are listed. At each succeeding level, the tasks from the previous one are further developed.

- 32.** Differentiate between a concrete step and an abstract step.

An abstract step is one in which further development is needed. A concrete step is one in which all the steps are fully specified.

- 33.** Write a top-down design for the following tasks.

Solutions are not unique.

- a.** Buying a toy for your four-year-old cousin.

*Go to store
Choose toy
Buy toy*

Go to store

*Choose store
Find location
Take bus*

Choose toy

*Walk up and down aisles
Panic at choices
Grab nearest large stuffed animal*

Buy toy

Go to clerk
Give stuffed animal to clerk
Give credit card to clerk
Sign credit card slip

b. Organizing an awards banquet for your soccer team.

Rent banquet room
Send invitations
Choose menu
Buy trophies

Rent banquet room

Find what is available
Visit possible choices
Choose one
Make reservation

Send invitations

Get list of people to invite
Buy invitations
Address invitations
Mail invitations

Buy trophies

Find out how many to buy
Find store that carries trophies
Order trophies over the phone
Pick up trophies

- c. Buying a dress or suit for an awards banquet at which you are being honored.

```
Go to favorite store
Choose dress or suit that suits you
Pay for choice
Go home
```

34. Write a top-down design for the following tasks.

Solutions are not unique.

- a. Calculating the average of ten test scores.

```
Set count to 0
Set sum to 0
While (count < 10)
    Get score
    Set sum to sum plus score
    Set count to count plus 1
Set average to sum divided by 10
```

- b. Calculating the average of an unknown number of test scores.

```
Set count to 0
Set sum to 0
While (there are more scores)
    Get score
    Set sum to sum plus score
    Set count to count plus 1
Set average to sum divided by count
```

- c. Describe the differences in the two designs.

The loop in the first design operates exactly 10 times. The loop in the second design operates as long as there are more scores.

35. Write a top-down design for the following tasks.

Solutions are not unique.

- a. Finding a telephone number in the phone book.

Find the right page
Find the right column
Search the column for name

Find the right page

Open to approximate part of book
While (page not found)
 Compare name with name on top of right page
 If (name on top is less)
 Turn page forward
 Else
 Compare name with name on top of left page
 If (name on top is greater)
 Turn page backward
 Else
 Page is found

Find right column

Current column is leftmost one
While (column not found)
 If (name on bottom of current column is greater)
 Column is found
 Else
 Set current column to one at right of current column

Search the column for name

```
Set found to false
While (more to look at and not found)
  Get next name
  If (name is the one you want)
    Get phone number
    Set found to true
If (found is false)
  Number not in book
```

- b.** Finding a telephone number on the Internet.

```
Log on to Internet
Go to favorite search engine
Type in "Find phone number"
Go to first response
Get phone number
Log off
```

- c.** Finding a telephone number on a scrap of paper that you have lost.

```
Search purses (wallets) for scrap of paper
Search waste paper baskets for scrap of paper
Search trash can for scrap of paper
```

Search purses (wallets)

```
While (paper not found and there are more purses or wallets)
  Get next one
  If (paper is there)
    paper is found
```

Search waste paper baskets

```

While (paper not found and there are more waste paper baskets)
  Get next one
  If (paper is there)
    paper is found

```

- d.** Describe the similarities and differences among these designs.

The first and third both have a process repeated a number of times; the second does not. The first and third are processes that most of us have done physically many times. The first and third involve a linear search through a container of data: columns in a book and purses (wallets), waste paper baskets, and trash cans.

- 36.** Distinguish between information and data.

Information is any knowledge that can be communicated. When information is in the form that a computer can use, it is called data. Thus data is any knowledge that can be communicated in a form that a computer can process.

- 37.** Write a top-down design for sorting a list of names into alphabetical order.

```

Repeat
  Scan list for name closest to beginning of the alphabet
    (smallest)
  Copy name to new list
  Cross name off original list
Until (There are no more names on original list)
Copy names back onto original list

```

- 38. a.** Why is information hiding important?

Information hiding defers details until the level where the details are important. This process keeps an algorithm from being dependent on the implementation details, which may change.

- b.** Name three examples of information hiding that you encounter every day.

Talking on the telephone.

Driving a car.

Turning on the television.

- 39.** An airplane is a complex system.

Solutions are not unique.

- a.** Give an abstraction of an airplane from the view of a pilot.

A pilot can view the airplane as a car that he or she drives on a highway of air.

- b.** Give an abstraction of an airplane from the view of a passenger.

A passenger can view the airplane as the inside of a limousine that is carrying the passenger from one place to another.

- c.** Give an abstraction of an airplane from the view of the cabin crew.

The cabin crew can view an airplane as a dining room.

- d.** Give an abstraction of an airplane from the view of a maintenance mechanic.

A maintenance mechanic can view an airplane as a collection of parts and wires put together according to his or her maintenance diagrams.

- e.** Give an abstraction of an airplane from the view of the airline's corporate office.

From the view of the boardroom, the airplane can be viewed as an expensive object used in the process of making money.

- 40.** List the identifiers and whether they named data or actions for the designs in Exercise 33.

- a.** Actions: go, choose, buy, find, give, sign

Data: store, toy, clerk, credit card

- b.** Actions: rent, send, choose, buy, find, visit, make, get, address, mail, order, pick up

Data: banquet room, invitations, menu, trophies, reservation, list of people, phone

- c.** Actions: go, choose, pay

Data: store, dress, suit, choice, home

- 41.** List the identifiers and whether they named data or actions for the designs in Exercise 34.

- a.** Actions: set, get

Data: count, sum, score, average

- b.** Actions: set, get

Data: count, sum, score, average

- 42.** List the identifiers and whether they named data or actions for the designs in Exercise 35.

- a.** Actions: find, search, open, compare, turn, set

Data: page, column, name, book, right page, left page

b. Actions: log on, go, type, get

Data: Internet, search engine, first response, phone number

43. Finish the verification of the address list example.

The next module to be verified is *Fill in missing data*. The first statement tells the user what to do if information is still not known. The next statement gets a name from the list. The third statement sets up a looping situation that continues as long as there are more names. How the expression on the *while* statement is evaluated is not our concern here. We assume we continue if there is more data. Within the loop, the last name is printed. The third loop statement asks if *street* is missing. We do not have to worry at this level how this is done, but true is returned if the data is missing. If the data is missing, the user is prompted to enter the data, and the data is read. We assume that the user enters the data correctly, thus the *street* is input correctly if that data is missing. The same logic can be applied to the next two *if* statements. The last statement gets another name from the list.

Put list in alphabetical order calls on a module to sort the list. We assume that the lower-level module is correct, so this module is correct.

Print the list writes a header label on the output. The next statement gets a name from the list. The third statement sets up a looping situation that continues as long as there are more names. The next four statements write the parts of the entry. We assume that the lower-level modules print the data correctly. A blank line is written to separate this person's data from the next. The last statement in the loop gets another name.

44. Take some sample data and desk check the address list example with actual values.

This is an activity, not a question.

45. Verify the designs in Exercise 33 using a walk-through.

This is an activity, not a question.

46. Verify the designs in Exercise 34 using an inspection.

This is an activity, not a question.

47. Verify the designs in Exercise 35 using top-down reasoning.

There are three tasks in the main module: *Find the right page*, *Find the right column*, *Search the column for the name*. If the first task is done correctly and the second task is done correctly, the third task has the proper column upon which to search. If the searching module works correctly, then the telephone number is found.

Module *Find the right page* starts by opening the phone book to the approximate page. A loop is set up to keep looking at pages until the correct one is found. This is done by comparing the name with the name on the top of the right-hand page, which is the “largest” name on the page. If the one for which we are searching is “larger,” then the right-hand page is turned and the process starts again. If the one for which we are searching is “smaller,” it is compared to the name on the top of the left-hand page, which is the “smallest” name on the page. If the name we are looking for is “smaller,” we turn the left-hand page. Otherwise, the name we are looking for is on the double page found.

Module *Find the right column*, sets the *current* column to the leftmost one. If the name on the bottom of the current column is “greater,” the right column is found; otherwise the current column is the next column on the right.

Module *Search the column for the name* sets *found* to false and starts looking name by name for the one whose phone number you want. The process of searching continues until the name is found or there are no more names to examine. If the name is found, the number is accessed. If the loop ends and *found* is still false, then the name was not in the book. If the loop expression is evaluated correctly and the comparison between names works correctly, then this module works correctly.

- b. There is only one module in this solution. The first three statements are actions, which we assume are correct. The fourth statement assumes that the first response from the search is an online telephone book. The fifth statement assumes that the number is in the phone book. Given these assumptions, the module is correct.

48. Distinguish between an object and an object class.

An object class is description of a group of objects with similar properties and behaviors. An object is a thing or entity that had meaning within a problem. An object is one of the things described by an object class.

49. Distinguish between a field and a method.

A field names data or actions within a class. A method is a named action within a class. Thus a field can contain a method.

50. How can objects relate to one another?

Objects can be related by containment, inheritance, or collaboration. An object can contain another object as a field. An object can inherit

the data and behavior of another object class. An object can collaborate with an object of its own class or another class.

- 51.** Discuss the differences between a top-down design and an object-oriented design.

Top-down design breaks the problem into successive levels of tasks; object-oriented design breaks the problem into successive levels of data objects.

- 52.** We outlined a strategy for developing an object-oriented decomposition.

- a.** List the four stages.

Brainstorming, filtering, scenarios, and responsibility algorithms.

- b.** Outline the characteristics of each stage.

Brainstorming is a group problem-solving activity that involves the spontaneous contribution of ideas from all members of the group. The output from this activity is a list of possible classes.

Filtering is a group activity in which the tentative list of classes is analyzed to determine if there are duplicates, if some classes share common attributes and behaviors, and if there are classes that really do not belong in the solution.

Scenarios are group activities that determine the responsibilities of the classes. They ask “what if” questions and determine if all possible situations have been considered.

Responsibility algorithms are the algorithms that implement the responsibilities. This phase is where the algorithms to carry out the solution get written.

- c.** What is the output from each of the four stages?

Brainstorming: a list of possible classes.

Filtering: CRC cards for the classes that survived this stage.

Scenarios: CRC cards with responsibilities outlined and collaborators indicated.

Responsibility algorithms: Algorithms for each of the responsibilities.

- d.** Are the stages independent? Explain.

No, each state is not independent. The first stage produces a tentative list that is used as input to the second stage. The second stage produces a list of classes that have survived the filtering stages as input to the third stage. The third stage produces completed CRC cards that are input to the fourth stage.

53. Design the CRC cards for an inventory system for a car dealership, using brainstorming, filtering, and scenarios.

Brainstorming: color, idnumber, date arrived at dealership, date sold, name of buyer, name of car, name of seller, date bought by dealer, date of last repair

Filtering: car, date, person, name, list of cars, idnumber, color

Scenarios:

Class Name: <i>Car</i>	Superclass:	Subclasses: <i>UsedCar</i>
Responsibilities	Collaborations	
<i>Initialize itself (name, date, color, idnumber)</i>	<i>String, Date, String, String</i>	
<i>Know color</i>	<i>String</i>	
<i>Know dateBought</i>	<i>Date</i>	
<i>Know name</i>	<i>String</i>	
<i>Know idnumber</i>	<i>String</i>	
<i>SetRepair (date)</i>	<i>Date</i>	
<i>KnowRepairDate</i>	<i>Date</i>	
<i>Sold (name, date)</i>	<i>Name, Date</i>	
<i>Know dateSold</i>	<i>Date</i>	
<i>Know buyer</i>	<i>Name</i>	
<i>Know repair</i>	<i>Date</i>	

Class Name: <i>UsedCar</i>	Superclass: <i>Car</i>	Subclasses:
Responsibilities	Collaborations	
<i>Initialize itself (car, date, name)</i>	<i>Car, Date, Name</i>	
<i>Know dateBought</i>	<i>Date</i>	
<i>Know fromWhom</i>	<i>Name</i>	

Class Name: <i>Date</i>	Superclass:	Subclasses:
Responsibilities	Collaborations	
<i>Initialize itself (month, day, year)</i>	<i>int, int, int</i>	
<i>Know month</i>	<i>int</i>	
<i>Know day</i>	<i>int</i>	
<i>Know year</i>	<i>int</i>	

Class Name: <i>Name</i>	Superclass:	Subclasses:
Responsibilities	Collaborations	
<i>Initialize itself (first, last)</i>	<i>String, String</i>	
<i>Know first</i>	<i>String</i>	
<i>Know last</i>	<i>String</i>	

Class Name: <i>List (from Library)</i>	Superclass:	Subclasses:
Responsibilities	Collaborations	
<i>Insert (car)</i>	<i>Car</i>	
<i>Delete (car)</i>	<i>Car</i>	
<i>.</i>		
<i>.</i>		
<i>.</i>		

54. Design the CRC cards for a database for a zoo using brainstorming, filtering, and scenarios.

Brainstorming: *family name, name, date of birth, date bought, food, cageNumber, sex, date of last shots*

Filtering: *animal, name, date, food, cageNumber, sex*

Scenarios:

Class Name: <i>Animal</i>	Superclass:	Subclasses:
Responsibilities	Collaborations	
<i>Initialize itself (familyName, name, dateOfBirth, dateBought, eatsWhat, cageNumber, sex)</i>	<i>String, String, Date, Date, String, String, Char</i>	
<i>Know familyName</i>	<i>String</i>	
<i>Know name</i>	<i>String</i>	
<i>Know food</i>	<i>String</i>	
<i>Know dateOfBirth</i>	<i>Date</i>	
<i>Know dateBought</i>	<i>Date</i>	
<i>Get shots (Date)</i>	<i>Date</i>	
<i>Know dateOfShots</i>	<i>Date</i>	
<i>Know cageNumber</i>	<i>String</i>	
<i>Know sex</i>	<i>Char</i>	
<i>Know dies</i>	<i>String</i>	

Class Name: <i>Date</i>	Superclass:	Subclasses:
Responsibilities	Collaborations	
<i>Initialize itself (month, day, year)</i>	<i>int, int, int</i>	
<i>Know month</i>	<i>int</i>	
<i>Know day</i>	<i>int</i>	
<i>Know year</i>	<i>int</i>	

Class Name: <i>List (from Library)</i>	Superclass:	Subclasses:
Responsibilities	Collaborations	
<i>Insert (animal)</i>	<i>Animal</i>	
<i>Delete (animal)</i>	<i>Animal</i>	
.		
.		
.		

55. Distinguish between data abstraction and procedural abstraction.

Data abstraction is the separation of the logical view of data from its implementation. Procedural abstraction is the separation of the logical view of an action from its implementation. Data abstraction is associ-

ated with data objects; procedural abstraction is associated with actions.

- 56.** What is a programming language?

A programming language is a sequence of instructions to perform a specified task written in a language that a computer can execute or that can be translated into a language that a computer can execute.

- 57.** Distinguish between syntax and semantics.

Syntax is the set of formal rules governing how instructions are written in a language. Semantics is the set of rules that give meaning to the instructions in a language.

Chapter 7 Exercises

For Exercises 1–15, mark the answers true and false as follows:

- A.** True
B. False

- 1.** Arithmetic can be performed in the Instruction Register.
B
- 2.** Arithmetic can be performed in the A Register.
A
- 3.** Arithmetic can be performed in the accumulator.
A
- 4.** The Z bit is 1 if the accumulator is zero.
A
- 5.** The N bit is 0 if the accumulator is negative.
B
- 6.** The Program Counter and the Instruction Register are two names for the same place.
B
- 7.** The A register and the accumulator are two names for the same place.
A
- 8.** The Instruction Register is three bytes long.
A
- 9.** The Program Counter is three bytes long.
B

10. The status bits are each one byte long.
B
11. The instruction specifier is two bytes long.
B
12. If the data to be loaded into the accumulator is stored in the operand, the instruction specifier is 00.
A
13. If the data in the accumulator is to be stored in the place named in the operand, the instruction specifier is 00.
B
14. All Pep/7 instructions occupy three bytes.
B
15. The branching instructions test the status bits.
A

Given the following state of memory (in hexadecimal), answer Exercises 16 through 20 by matching the problem to the solution shown.

0001 A2

0002 11

0003 FF

0004 00

A. 10100010 00010010

B. 11111111 00000000

C. 00000000 00000011

D. 11101101 00000001

E. 00010010 00000000

16. What are the contents of the A register after the execution of this instruction?

00001000 00000000 00000011

C

17. What are the contents of the A register after the execution of this instruction?

00001001 00000000 00000011

B

- 18.** What are the contents of the A register after the execution of the following two instructions?

```
00001001 00000000 00000001
00011000 00000000 00000001
```

A

- 19.** What are the contents of the A register after the execution of the following two instructions?

```
00001000 00000000 00000001
00011001 00000000 00000010
```

E

- 20.** What are the contents of the A register after the execution of the following two instructions?

```
00001001 00000000 00000011
00100001 00000000 00000010
```

D

Exercises 21–53 are short answer or programs.

- 21.** What does it mean when we say that a computer is a *programmable* device?

Programmable means that data and instructions are logically the same and are stored in the same place. The consequence of this fact is that the program the computer executes is not wired into the hardware but entered from outside.

- 22.** List five operations that any machine language must include.

There must be machine-language instructions to store, retrieve, and process data, to input data, and to output data. These instructions mirror the operations of the von Neumann machine.

- 23.** The distinction between concrete and abstract steps in algorithms is not always clear cut. Discuss this dilemma and give concrete examples to support your discussion.

Algorithms are eventually coded in a programming language. Different programming languages represent different levels of abstraction. What takes a single step in one language may take many steps in another language. Thus, a concrete step in one language may be an abstract step in another.

- 24.** What is a virtual machine? Discuss this definition in terms of the Pep/7 computer.

A virtual machine is a hypothetical machine designed to illustrate important features of a real computer. The Pep/7 computer is a virtual machine designed to illustrate the features of the von Neumann architecture. It has instructions to store, retrieve, and process data as well as instructions to input and output data.

- 25.** We said that you should have guessed that a Pep/7 instruction would use 5 bits when we said that there were 32 instructions. Explain.

It takes exactly 5 bits to represent 32 different things, instructions in this case.

- 26.** Describe the features of the Pep/7 CPU that we covered in this chapter.

There is one register for arithmetic and logical operations: the A register (the accumulator). There is a Program Counter that contains the address of the next instruction to be executed and the Instruction Register that contains the instruction being executed. There are two status bits, N and Z, which are one if the accumulator is negative or zero if the accumulator is positive, respectively. Memory is byte addressable.

- 27.** We covered only two of the four addressing modes. If we had not stated this explicitly, could you have deduced that this was true? Explain.

If there were only two addressing modes, one bit would have been used instead of two. Because two bits are used, there must be four modes.

- 28.** Where is the data (operand) if the address mode specifier is

a. 00

If the address mode specifier is 00, the data is in the operand specifier.

b. 01

If the address mode specifier is 01, the data is stored in the place named in the operand specifier.

- 29.** Distinguish between the IR (instruction register) and the PC (program counter).

The IR contains an instruction (the one being executed); the PC contains an address (the address of the next instruction to be executed).

- 30.** How many bits are required to address the Pep/7 memory?

The Pep/7 memory contains 4096 bytes, so 12 bits are required to address each one.

- 31.** How many more cells could be added to memory without having to change the instruction format? Justify your answer.

The operand specifier is 16 bits long. Therefore 2^{16} different bytes could be addressed without changing the instruction format. Thus, 61440 more bytes could be added.

- 32.** Some Pep/7 instructions are unary, taking only one byte. Other instructions require three bytes. Given the instructions that we have covered in this chapter, would it be useful to define instructions that require only two bytes?

The instructions we have covered, other than the Stop instruction, use the operand specifier of the instruction. The operand specifier is two bytes long, so three bytes are required for the instruction: the instruction specifier and the operand specifier. Therefore, two-byte instructions would not be useful.

- 33.** If the input character is A, what is the result of executing the following two instructions?

```
0001 11011001 00000000 00000110
```

```
0004 11100000 00000000 00001010
```

A is written on the screen.

- 34.** If the input character is A, what is the result of executing the following two instructions?

```
0001 11011001 00000000 00000110
```

```
0004 11100001 00000000 00001010
```

What ever is stored in location 01000001 is written on the screen.

- 35.** Write the algorithm for writing out your name, given that the implementation language is Pep/7 machine code.

```
Write "Nell"
```

Write "Nell"

```
Write "N"
```

```
Write "e"
```

```
Write "l"
```

```
Write "l"
```


Write “N”

Write 4E (hex)

Write “e”

Write 65 (hex)

Write “l”

Write 6C (hex)

Write “l”

Write 6C (hex)

- 36.** Write the machine-language program to implement the algorithm in Exercise 35.

E0 00 4E E0 00 65 E0 00 6C E0 00 6C zz

- 37.** Write the algorithm for writing out your name, given that the implementation language is Pep/7 assembly language.

Write “Nell”

Write “Nell”

Write “N”

Write “e”

Write “l”

Write “l”

- 38.** Write the assembly-language program to implement the algorithm in Exercise 37.

```
CHARO h#0010,d ;Output 'N'
CHARO h#0011,d ;Output 'e'
CHARO h#0012,d ;Output 'l'
CHARO h#0013,d ;Output 'l'
STOP
.ASCII /Nell/ ;Store 'Hello' into proper places
.END
```

- 39.** Rewrite the machine language program in 7.4, using direct addressing.

```
E1 00 10 E1 00 11 E1 00 12 E1 00 13 E1 00 14 00 48 65 6C
6C 6F zz
```

- 40.** Distinguish between the Pep/7 menu options *Load*, *Load/Execute*, and *Execute*.

Load puts the program in memory ready to execute. Load/Execute loads the program into memory and executes it. If a program is loaded, an Execute command executes the program. If the active window contains a machine language program and there is no loaded program, the Execute command loads and executes it.

- 41.** The following program seems to run, but does strange things with certain input values. Can you find the bug?

```
BR Main
sum: .WORD d#0
num1: .BLOCK d#1
num2: .BLOCK d#1
num3: .BLOCK d#1
Main: LOADA sum,d
      DECI num1,d
      DECI num2,d
      DECI num3,d
      ADDA num3,d
      ADDA num2,d
      ADDA num1,d
      STOREA sum,d
      DECO sum,d
      STOP
      .END
```

One byte of storage is set up for each input value. If the value that is read is greater than one byte, the excess spills over to the byte above, giving the wrong answer.

- 42.** Correct the code in Exercise 41 and run the test plan outlined in the chapter.

```
BR    Main
sum:   .WORD d#0
num1:  .BLOCK d#2
num2:  .BLOCK d#2
num3:  .BLOCK d#2
Main:  LOADA sum,d
       DECI num1,d
       DECI num2,d
       DECI num3,d
       ADDA num3,d
       ADDA num2,d
       ADDA num1,d
       STOREA sum,d
       DECO sum,d
       STOP
       .END
```

The test plan gives the correct answers with this code.

- 43.** Finish executing the test plan for the algorithm in the text that reads and sums three values.

The answers are correct.

- 44.** Write an algorithm that reads in three values and writes out the result of subtracting the second value from the sum of the first and the third values.

```
Read num1
Read num2
Read num3
Load num1
Add num3
Sub num2
Store in answer
Write answer
```

45. Implement the algorithm in Exercise 44 as an assembly-language program.

```
BR    Main
answer: .WORD d#0
num1: .BLOCK d#2
num2: .BLOCK d#2
num3: .BLOCK d#2
Main: DECI num1,d
      DECI num2,d
      DECI num3,d
      LOADA num1,d
      ADDA num3,d
      SUBA num2,d
      STOREA answer,d
      DECO answer,d
      STOP
      .END
```

46. Write and implement a test plan for the program in Exercise 45.

Reason for Test Case	Input Values	Expected Output	Observed Output
Assumption: Input values are no greater than $2^{15} - 1$ or less than -2^{15}			
Input 3 positive numbers	4, 6, 1	-1	-1
Input 3 negative numbers	-4, -6, -1	1	1
Input mixed numbers	4, 6, -1	-3	-3
	4, -6, 1	11	11
	4, -6, -1	-9	-9
	-4, 6, 1	-9	-9
	-4, 6, -1	-11	-11
	-4, -6, 1	3	3
Large numbers	32767, -1, +1, 32767, 1, -1	overflows 32767	overflows

47. Design and implement an algorithm in assembly language that reads four values and prints the sum.

```
Read num1
Read num2
Read num3
Read num4
Load num1
Add num2
Add num3
Add num4
Store in answer
Write answer
```

```
BR    Main
answer: .WORD d#0
num1:  .BLOCK d#2
num2:  .BLOCK d#2
num3:  .BLOCK d#2
num4:  .BLOCK d#2

Main:  DECI  num1,d
       DECI  num2,d
       DECI  num3,d
       DECI  num4,d
       LOADA num1,d
       ADDA  num2,d
       ADDA  num3,d
       ADDA  num4,d
       STOREA answer,d
       DECO  answer,d
       STOP
       .END
```

- 48.** Is the test plan for a machine language program valid for the same solution written in assembly language? Explain your answer.

A data-coverage plan is written without looking at the code, so the same test plan would be valid. A code-coverage plan looks at the code, but because there is a one-to-one relationship between a machine code instruction and an assembly language instruction, the same test plan would be valid.

- 49.** Distinguish between the pseudocode instructions `.BLOCK` and `.WORD`.

The pseudocode instruction `.BLOCK` takes a decimal argument and sets aside that many bytes of storage and sets them to zero. A `.WORD` pseudocode instruction takes a decimal argument and generates one word of storage with the decimal value stored in it.

- 50.** Distinguish between assembly language pseudocode instructions and mnemonic instructions.

Pseudocode instructions are instructions to the assembler; mnemonic instructions are to be translated by the assembler.

- 51.** Distinguish between test plans based on code coverage and data coverage.

A code-coverage test plan is based on examining and covering the statements in the code. A data-coverage test plan is based on the input data to the program.

- 52.** Explain the meaning of the Pep/7 menu option `Execution Input`.

The `Execution Input` menu asks if you want the input data to come from the current window or from the keyboard.

- 53.** Write the Pep/7 assembly language statement for the following instructions.

- a.** Branch to location `Branch1` if the accumulator is zero.

`BREQ Branch1`

- b.** Branch to location `Branch1` if the accumulator is negative.

`BRLT Branch1`

- c.** Branch to location `Branch1` if the accumulator is negative and to `Branch2` if accumulator is not negative.

`BRLT Branch1`

`Branch2:`

That is, go to location `Branch1` if the accumulator is negative. If the accumulator is not negative, the next instruction is executed, so it must be labeled `Branch2`.

Chapter 8 Exercises

For Exercises 1–14, match the question with the appropriate translation or execution system.

- A.** Interpreter
- B.** Assembler
- C.** Compiler
- D.** Machine code

1. What translates a high-level language into machine code?
C
2. What translates a Java program into Bytecode?
C
3. What executes Bytecode?
A
4. What translates an assembly language program?
B
5. What is the output of an assembler?
D
6. What takes input in a high-level language and directs the computer to perform the actions specified in each statement?
A
7. What executes the Java Virtual Machine?
D
8. What is used to translate a program in ALGOL?
C
9. What is used to translate a program in APL?
A
10. What is used to translate a program in COBOL?
C
11. What is used to translate a program in FORTRAN?
C
12. What is used to translate a program in Lisp?
A
13. What is used to translate a program in SNOBOL4?
A
14. Which translator runs the slowest?
A

For Exercises 15–36, match the language paradigm and the language or language description.

- A. Imperative or procedural
- B. Functional
- C. Logic
- D. Object oriented

- E. Procedural language with some object-oriented features
- F. Object-oriented language with some procedural features

15. Which paradigm most accurately describes FORTRAN?

A

16. Which paradigm most accurately describes C++?

E

17. Which paradigm most accurately describes PASCAL?

A

18. Which paradigm most accurately describes Java?

F

19. Which paradigm most accurately describes Lisp?

B

20. Which paradigm most accurately describes BASIC?

A

21. Which paradigm most accurately describes PROLOG?

C

22. Which paradigm most accurately describes SIMULA?

D

23. Which paradigm most accurately describes ALGOL?

A

24. Which paradigm most accurately describes ML?

B

25. Which paradigm most accurately describes Scheme?

B

26. Which paradigm most accurately describes Ada?

A

27. Which paradigm most accurately describes C?

A

28. Which paradigm most accurately describes Smalltalk?

D

29. The dominant languages used in industry throughout the history of computing software come from which paradigm?

A

30. Which paradigm did the Japanese choose for the fifth-generation computer?

C

- 31.** Which paradigm allows the programmer to express algorithms as a hierarchy of objects?
D
- 32.** Which paradigm allows the programmer to express algorithms as a hierarchy of tasks?
A
- 33.** Which paradigm allows the programmer to express algorithms as mathematical functions?
B
- 34.** Which paradigm has no assignment statement?
B
- 35.** Which paradigm uses recursion exclusively to express repetition?
B
- 36.** Which paradigm has no variables?
B

Exercises 37–82 are short answer.

- 37.** What is the hallmark of an assembly language?
The hallmark of an assembly language is that each assembly language instruction is translated into one machine language instruction.
- 38.** Distinguish between an assembler and a compiler.
An assembler translates assembly-language instructions into machine code. A compiler translates high-level language instructions into machine code. The translation of an assembler is one to one: One statement in assembly language is translated into one statement in machine code. The translation of a compiler is one to many: One high-level language instruction is translated into many machine language instructions.
- 39.** Distinguish between a compiler and an interpreter.
The output from a compiler is a machine-language program. That program may be stored for later use or immediately executed, but the execution is a distinct process from the translation. The output from an interpreter is a solution to the original problem, not a program that when executed gives you the solution.
- 40.** Compare and contrast an assembler, a compiler, and an interpreter.
All three are translators. They differ in the complexity of the languages they translate and in the output from the translator. Assemblers and compilers produce machine-language programs, which when

run, solve the original problem; interpreters produce solutions to the original problem. Assemblers translate very simple languages; compilers and interpreters can translate very complex languages.

- 41.** Describe the portability provided by a compiler.

A program written in a high-level language that is compiled can be translated and run on any machine that has a compiler for the language.

- 42.** Describe the portability provided by the use of Bytecode.

Bytecode is the output from (usually) a Java compiler. There is a virtual machine for which Bytecode is the machine language. A program compiled into Bytecode can be executed on any system that has a simulator for the virtual machine. The Java Virtual Machine (JVM) executed Bytecode.

- 43.** Describe the process of compiling and running a Java program.

A Java program is compiled into Bytecode, which can be executed on any system with a JVM.

- 44.** Discuss the word *paradigm* as it relates to computing.

Programming languages reflect differing views of reality, which we call paradigms. We use these views (paradigms) to classify the languages.

- 45.** Name four programming language paradigms and give an example language in each.

Imperative or procedural paradigm: Fortran, Basic, Cobol, C, C++

Functional paradigm: Lisp, Scheme, ML, FP

Logic paradigm: PROLOG

Object-oriented paradigm: Simula, Smalltalk, Java

- 46.** What are the characteristics of the imperative paradigm?

Programs describe the processes necessary to solve the problem.

- 47.** What are the characteristics of the functional paradigm?

Programs are expressed as the evaluation of functions.

- 48.** What are the characteristics of the logic paradigm?

Rules of logic are used to deduce answers from facts and rules.

- 49.** How does the view of an object-oriented program differ from the view of an imperative program?

An object-oriented view of a program is that of interacting objects; the imperative view of a program is that of interacting tasks.

- 50.** How do you ask questions in a programming language?

To ask a question in a programming language, you make an assertion. If the assertion is true, the answer is true. If the assertion is false, the answer is false.

- 51.** What is a Boolean variable?

A Boolean variable is a place in memory, referenced by an identifier, that can contain true or false.

- 52.** What is a Boolean expression?

A Boolean expression is a sequence of identifiers, separated by compatible operators, that evaluates to true or false.

- 53.** Given Boolean variables *one*, *two*, and *three*, write an assertion for each of the following questions.

- a.** Is one greater than both two and three?

$(one > two) \text{ AND } (one > three)$

- b.** Is one greater than two, but less than three?

$(one > two) \text{ AND } (one < three)$

- c.** Are all three variables greater than zero?

$(one > 0) \text{ AND } (two > 0) \text{ AND } (three > 0)$

- d.** Is one less than two or one less than three?

$(one < two) \text{ OR } (one < three)$

- e.** Is two greater than one and three less than two?

$(two > one) \text{ AND } (three < two)$

- 54.** Write the operation table for Boolean operation AND.

AND	0	1
0	0	0
1	0	1

- 55.** Write the operation table for Boolean operation OR.

OR	0	1
0	0	1
1	1	1

56. Write the operation table for Boolean operation NOT.

NOT	
0	1
1	0

57. What is a data type?

A data type is the description of a set of values and the basic set of operations that can be applied to values of the type.

58. What is strong typing?

Strong typing means that each variable is assigned a data type and only values of that type can be stored in the variable.

59. Define the following data types.

- a. integer

The range of integer values that a machine can represent.

- b. real

The range of rational numbers that a machine can represent.

- c. character

The character in the character set that the machine supports.

- d. Boolean

The values true and false.

60. Is the string data type an atomic data type? Justify your answer.

A string can be output like an atomic data type, but it is made up of characters, each of which can be accessed separately. Thus, you can argue either side.

61. If the same symbol is used for both single characters and strings, how can you distinguish between a single character and a one-character string?

If the same symbol is used, a character cannot be distinguished from a one-character string.

62. What is a declaration?

A declaration is an instruction to the compiler that associates an identifier with a variable, an action, or some other entity within the language that can be given a name. The programmer can then refer to that entity by name.

63. Fill in the following table showing the appropriate syntactic marker or reserved word for the language shown based on your observation of the table on page 239.

Language	Ada	VB.NET	C++	Java
Comments	--	'	//	//
End of statement	;	end of line or comment	;	;
Assignment statement	:=	=	=	=
Real data type	Float	Single	float	float
Integer data type	Integer	Integer	int	int
Beginning of declaration(s)	none	Dim	none	none

64. How do the .WORD and .BLOCK assembler directives in the Pep/7 assembly language differ from the declarations in high-level languages?

.WORD allows us to associate an identifier with a word in storage and specify what should go into that word. .BLOCK allows us to associate an identifier with a specified number of bytes. In each case the programmer knows the actual address associated with the identifier. Declarations in high-level languages give you the same functionality, but the programmer does not need to worry about the actual address.

65. Distinguish between instructions to be translated and instructions to the translating program.

Instructions to the translating program tell the program to associate identifiers with objects, and if the objects are data, tell the program the data type of what can be stored in that place.

66. Consider the following identifiers: *Address*, *ADDRESS*, *AddRes*, *Name*, *NAME*, *Name*

- a. How many different identifiers are represented if the language is Ada?

2

- b. How many different identifiers are represented if the language is VB.NET?

6

- c. How many different identifiers are represented if the language is C++ or Java?

6

- 67.** Explain the operation of the sequence control structure.
The sequence control structure says to execute the statements in sequence, one after the other.
- 68.** Explain the flow of control of the *if* statement.
If the Boolean expression is true, execute the first statement. If the Boolean expression is false, execute the second statement. In either case, continue with the statement following the second statement.
- 69.** How does a *case* statement differ from an *if* statement?
An *if* statement is based on an expression being true or false. A *case* statement matches the contents of an expression with a series of values to find the statement to execute next.
- 70.** What is the flow of control in a *while* statement?
The statements within the *while* statement are repeated as long as the *while* expression is true. When the expression becomes false, the statement following the end of the loop body is executed.
- 71.** What is recursion?
Recursion is the ability of a subprogram to call itself.
- 72.** How does recursion act as a repetition structure?
The program uses a selection statement to determine whether to stop the process or to repeat the code by calling the subprogram again.
- 73.** A looping structure uses a while (loop) statement; a recursive structure uses a if (selection) statement.
- 74.** Explain the statement, “Subprograms are a powerful tool for abstraction.”
A subprogram is a named task. The calling program can be designed and written using the subprogram name without knowing how the subprogram is implemented.
- 75.** Describe how a parameter list is used to communicate information from the calling unit to the subprogram.
A parameter list is a list of identifiers, separated by commas. These identifiers are just place holders. When the subprogram is called, the calling program sends the actual data the subprogram is to use. The variables in the argument list replace the identifiers in the parameter list.

- 76.** Distinguish between a parameter and an argument.

A parameter is a dummy name listed on the subprogram's heading. An argument is the data the calling program sends to the subprogram to use.

- 77.** Distinguish between a value parameter and a reference parameter.

The calling program sends a copy of the data to the subprogram if the parameter is a value parameter. If the subprogram's parameter is a reference parameter, the calling program sends the address of the data.

- 78.** What three elements must be present in the definition of a record?

The record name, the names of the individual data items, and their respective data types.

- 79.** Ada uses a range of index values to define an array, but VB.NET and C++ specify the number of places in the array. Explain.

Ada lets the user explicitly define how the values in the array are to be indexed; the number of items is determined from this range. VB.NET and C++ specify the number of elements in the array, and accessing is always from [0] through [number of elements – 1]

- 80.** Examine the following three array declarations:

```
type Index is range -1..10; -- Ada
type Data_Array is array (Index) of Integer;
Data : Data_Array;          -- Ada
```

```
Dim data(11) As Integer      ' VB.NET
int data[11];                // C++
```

Are the arrays declared the same? Justify your answer.

The Ada array contains 12 slots, indexed from (-1)..(10).

The VB.NET and C++ arrays contain 11 slots, indexed from (0)..(10) in VB.NET and [0]..[10] in C++.

- 81.** Distinguish between the definition of an object in the design phase and in the implementation phase.

An object in the design phase is an entity that has meaning within the context of the problem. An object in the implementation phase is an instance of a class.

- 82.** Distinguish between the definition of a class in the design phase and in the implementation phase.

A class in the design phase is a description of a group of objects with similar properties and behaviors. A class in the implementation phase is a pattern for an object.

Chapter 9 Exercises

For Exercises 1–10, match the class of operation with the operation described.

- A.** Constructor: Creates a new instance of a container.
- B.** Transformer: Changes the contents of a container.
- C.** Iterator: Allows access to each item in a container one at a time.
- D.** Observer: Asks information about a container without changing it.

1. Put an item into a container.

B

2. Create a new container.

A

3. How many objects are in the container.

D

4. Is the container empty?

D

5. Get the next item in the container.

C

6. Have we looked at all the items in the container?

D

7. Delete an item from the container.

B

8. Is *item* in the container?

D

9. Set the container to empty.

B

10. Is the container full?

D

For Exercises 11–20, match the type of list implementation with the given implementation step.

- A.** Array-based
- B.** Linked
- C.** Either
- D.** Neither

11. Initialize by setting *length* to zero.

A

- 12. In an unsorted container, put item into the beginning of the list.
B
- 13. In a sorted container, put the item at the end of the list.
D
- 14. To insert in a sorted list, first find the appropriate position.
C
- 15. Initialize by setting pointer to *null*.
B
- 16. *Get next item* involves incrementing a counter.
A
- 17. *Get next item* involves moving a pointer.
B
- 18. *More items* involves comparing length to a counter.
A
- 19. *More items* involves comparing something to null.
B
- 20. Find the item involves searching the list.
C

For Exercises 21–37, mark the answers true and false as follows:

- A. True
- B. False
- 21. A binary search cannot be applied to a linked list.
A
- 22. A linear search cannot be applied to an array-based list.
B
- 23. A binary search cannot be applied to an array-based list.
B
- 24. A binary search is always faster than a linear search.
B
- 25. A binary search cannot be applied to an unsorted list.
A
- 26. A stack exhibits LIFO behavior.
A
- 27. A queue exhibits FIFO behavior.
A

- 28.** A stack and a queue are different versions of the same abstract data type.
B
- 29.** A binary search tree allows $\log_2 N$ searching in a linked structure.
A
- 30.** In a graph, the vertices represent the objects being modeled.
A
- 31.** The bubble sort algorithm involves finding the smallest item in the unsorted portion of the array and swapping it with the current item.
B
- 32.** The selection sort algorithm involves finding the smallest item in the unsorted portion of the array and swapping it with the current item.
A
- 33.** The bubble sort algorithm swaps every out-of-order pair it sees.
A
- 34.** The Quicksort algorithm swaps every out-of-order pair encountered from different ends of the array.
A
- 35.** The Quicksort algorithm is always quick.
B
- 36.** The shape of a binary search tree depends on the order in which the items are inserted.
A
- 37.** The edges in a graph represent relationships.
A

Exercises 38–61 are problems or short answer questions.

- 38.** *Abstract data types, data structures, and containers:*

a. Define these terms

Abstract data types are data types whose properties (domains and operations) are specified independently of any particular implementation.

A data structure is the implementation of the composite data fields of an abstract data type.

A container is an object whose role is to hold and manipulate other objects.

- b. What do they have in common?
Each represents the concept of collections of data objects.
 - c. What distinguishes each from the others?
Each represents a different level. An ADT is the logical view of the properties of a class of data. A data structure is the implementation level of this logical view. A container is the description given to all logical views of this type of object; it represents how an application program might view the ADT at a higher level.
- 39.** Name and describe the three views of data.
Application, logical, and implementation are the three views of data. The application is the view of the data within a particular problem. The logical is the view that sees the data as objects with similar properties and behaviors and specifies them at an abstract level. The implementation view is a specific representation of the data and the actions to manipulate them.
- 40.** *Array-based implementation and linked implementation:*
- a. Define these terms.
An array-based implementation is one that uses an array to hold the data items. Each item is accessed by its place in the structure. A linked implementation is one that uses a structure made up of nodes in which the nodes are explicitly linked one to another.
 - b. What do they have in common?
They are both storage structures to hold collections of homogeneous items.
 - c. What distinguishes one from the other?
They are distinguished by how the items are stored. In the array-based implementation, data items are stored in contiguous locations in memory and accessed by position. In a linked implementation, data items are stored in nodes along with the information of where to find the next node in the structure.
- 41.** Draw the unsorted list containing the following strings: *blue, black, green, yellow, red, purple, white, and violet.*
- a. In an unsorted array-based list.
 - b. In a sorted array-based list.
 - c. In an unsorted linked list.
 - d. In a sorted linked list.

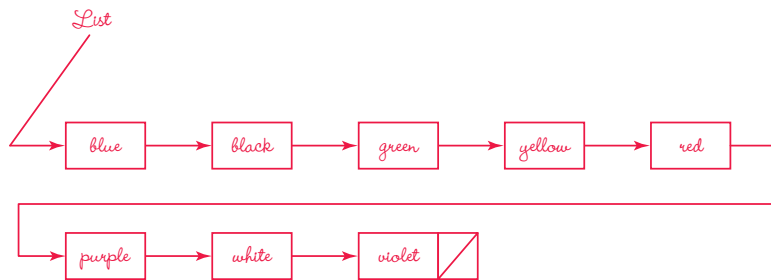
- a. An unsorted array-based list.

8	blue	[0]
	black	[1]
	green	[2]
	yellow	[3]
	red	[4]
	purple	[5]
	white	[6]
	violet	[7]

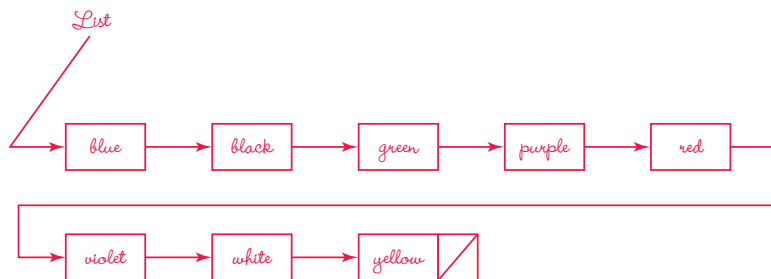
- b. A sorted array-based list.

8	black	[0]
	blue	[1]
	green	[2]
	purple	[3]
	red	[4]
	violet	[5]
	white	[6]
	yellow	[7]

- c. An unsorted linked list.



- d. A sorted linked list.



42. Give the meaning of the following expressions in an array-based implementation:

- a. *Put item*

Put item means that given an index, shift the items that follow down one slot in the array and store the item at the index position.

- b. *Remove the item*

Remove the item means that given an index, shift the items that follow up one slot in the array.

- c. *Get next item*

Get next item means to increment the value used as an index and access that indexed position.

- d. *More items?*

More items means that the variable used as an index is less than $\text{length} - 1$.

- 43.** Give the meaning of the following expressions in a linked implementation:
- Put item*
Put item means given *current*, insert a new node with *item* in the info part of the node between *current* and *next(current)*.
 - Remove the item*
Remove the item means given *current*, remove the *next(current)*.
 - Get next item*
Get next item means to set *current* to *next(current)*.
 - More items*
More items means that *current* does not contain null.

Questions 44 through 46 use the following list of values.

length	list	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]
11		23	41	66	20	2	90	9	34	19	40	99

- 44.** Show the state of the list when *current* is first set equal to the 4th item in the selection sort.

Array when current is first set to 4th item.

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]
2	9	19	20	23	90	41	34	66	40	99

- 45.** Show the state of the list when *current* is first set equal to the 5th item in the bubble sort algorithm.

Array when current is first set equal to the 5th item.

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]
2	9	19	20	23	41	66	34	40	90	99

- 46.** Show the state of the list when the first recursive call is made in Quicksort using list[0] as split value.

Array when first recursive call is made.

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]
2	19	9	20	23	90	66	34	41	40	99

Questions 47 through 48 use the following list of values.

length	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]
11	5	7	20	33	44	46	48	99	101	102	105

- 47.** How many comparisons does it take using a sequential search to find the following values or determine that the item is not in the list?
- a. 4
 - 11
 - b. 44
 - 5
 - c. 45
 - 11
 - d. 105
 - 11
 - e. 106
 - 11
- 48.** How many comparisons does it take using a binary search to find the following values or determine that the item is not in the list?
- a. 4
 - 4
 - b. 44
 - 4
 - c. 45
 - 4
 - d. 105
 - 4
 - e. 106
 - 4
- 49.** Write the algorithm for *Enque* in an array-based implementation.
There are several solutions. The easiest is to keep the front of the queue always in the [0] position.

```
// rear is index of last element put on the structure
Increment rear
Set items[rear] to newItem
```



50. Write the algorithm for Pop in a linked implementation.

51. Write the algorithms for Deque in a linked implementation.

Exercises 52–57 use the following tree.



52. Name the content of each of the leaf nodes.

alex, christopher, john, robert, june, kit, nell, susy

53. List the contents of each node that has just a right child.

al, chris, mari, sarah

54. List the contents of each node that has just a left child.

There are none.

55. What is the height of the tree?

5

56. Name the content of the nodes that are the ancestors of the node containing *kit*.

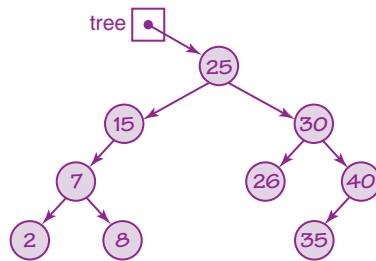
kate, lila, phil, john

57. How many descendants does the node containing *jim* have?

3

58. Draw the tree that results from inserting the following values in this order.

25, 15, 7, 30, 26, 8, 2, 40, 35



Chapter 10 Exercises

For Exercises 1–18, mark the answers true and false as follows:

A. True

B. False

1. An operating system is an example of application software.
B
2. An operating system provides a basic user interface that allows the user to use the computer.
A
3. A computer can have more than one operating system, but only one is in control at any given time.
A

4. Multiprogramming is the technique of using multiple CPUs to run programs.
B
5. In the 1960s and 70s, a human operator would organize similar computer jobs into batches to be run.
A
6. Batch processing implies a high level of interaction between the user and the program.
B
7. A timesharing system allows multiple users to interact with a computer at the same time.
A
8. A dumb terminal is an I/O device that connects to a mainframe computer.
A
9. A logical address specifies an actual location in main memory.
B
10. An address in a single contiguous memory management system is made up of a page and an offset.
B
11. In a fixed partition system, main memory is divided into several partitions of the same size.
B
12. The bounds register contains the last address of a partition.
B It contains the length of the partition.
13. The first page in a paged memory system is page 0.
A
14. A process in the running state is currently being executed by the CPU.
A
15. The process control block (PCB) is a data structure that stores all information about a process.
A
16. CPU scheduling determines which programs are in memory.
B
17. The first-come, first-served scheduling algorithm is provably optimal.
B

- 18.** A time slice is the amount of time each process is given before being preempted in a round robin scheduler.

A

For Exercises 19–23, match the operating system with information about it.

- A.** MacOS
- B.** Unix
- C.** Linux
- D.** DOS
- E.** Windows

- 19.** Which is the operating system of choice for Apple Computers?

A.

- 20.** Historically, which is the operating system of choice for serious programmers?

B

- 21.** Which is the PC version of Unix?

C

- 22.** What is the Microsoft operating system family provided on PCs called?

E

- 23.** What is the original PC operating system called?

D

For Exercise 24–26, match the following software type with its definition.

- A.** Systems software
- B.** Operating system
- C.** Application software

- 24.** Programs that help us solve real-world problems.

C

- 25.** Programs that manage a computer system and interact with hardware.

A

- 26.** Programs that manage computer resources and provide an interface for other programs.

B

Exercises 27–75 are problems or short answer questions.

- 27.** Distinguish between application software and system software.

Systems software are tools to help others write programs; they manage a computer system and interact with hardware. Application software are programs to solve specific problems.

- 28.** What is an operating system?

An operating system is a piece of software that manages a computer's resources and provides an interface for system interaction.

- 29.** Explain the term *multiprogramming*.

Multiprogramming is the technique of keeping multiple programs in main memory at the same time, each competing for time on the CPU.

- 30.** The following terms relate to how the operating system manages multiprogramming. Describe the part each plays in this process.

- a.** Process

A process is a program in execution.

- b.** Process management

Process management is keeping track of necessary information for active processes.

- c.** Memory management

Memory management is keeping track of how and where programs are loaded into main memory.

- d.** CPU scheduling

CPU scheduling is determining which process in memory is given access to the CPU so that it may execute.

- 31.** What constitutes a batch job?

A batch job was made up of the program and the instructions regarding the system software and other resources needed to execute the job.

- 32.** Describe the evolution of the concept of batch processing from the human operator in the 1960s and 70s to the operating systems of today.

Originally the instructions regarding the system software needed for a program were given to the human operator. Today the instructions are given directly to the computer through OS commands that are part of the file containing the program. Today, batch processing has come to mean a system in which programs and system resources are coordinated and executed without interaction between the user and the program.

- 33.** Define *timesharing*.

Timesharing is a technique by which CPU time is shared among multiple interactive users at the same times.

- 34.** What is the relationship between multiprogramming and timesharing?

Multiprogramming allows multiple processes to be active at once. Timesharing allows the multiple processes to be interactive ones.

- 35.** Why do we say that users in a timesharing system have their own virtual machine?

Users have the illusion of having the computer all to themselves.

- 36.** In Chapter 7, we defined a virtual machine as a hypothetical machine designed to illustrate important features of a real machine. In this chapter, we define a virtual machine as the illusion created by a timesharing system that each user has a dedicated machine. Relate these two definitions.

The illusion created in a timesharing situation is that the user owns a single hypothetical machine. The hypothetical machine illustrates the important features of the single machine the user needs.

- 37.** How does the timesharing concept work?

Each user is represented by a login process that runs on the mainframe. When the user runs a program, a new process is created that competes for CPU time with other processes. The rationale is that the computer is so fast that it can handle multiple users without anyone having to wait.

- 38.** What is a *real-time system*?

A real-time system is a system in which the speed of an answer is crucial.

- 39.** What is *response time*?

Response time is how long it takes to get an answer. The expression comes from the delay between receiving a stimulus (asking a question) and producing a response (answering the question).

- 40.** What is the relationship between real-time systems and response time?

Time is critical in many real-time situations, so the response time must be kept to a minimum.

- 41.** In a multiprogramming environment, many processes may be active. What are the tasks that the OS must accomplish in order to manage the memory requirements of active processes?

The OS must keep track of where and how a program resides in memory and convert logical program addresses into actual memory addresses.

- 42.** Distinguish between logical addresses and physical addresses.

A physical address is an actual address in the computer's main memory device. A logical address is an address relative to the program. A logical address is sometimes called a relative address for obvious reasons.

- 43.** What is *address binding*?

Address binding is the mapping of a logical address into a physical address.

- 44.** Name three memory-management techniques and give the general approach taken in each.

Single contiguous memory management: Only the OS and one application program are loaded into memory at the same time.

Static and dynamic partitions: More than one program is loaded into memory with the OS at the same time. Each application program is given its own partition of memory.

Paging: Main memory is divided into fixed-sized blocks called frames, and processes are divided into fixed-sized blocks called pages. Any number of programs can be loaded with the OS, but a process does not necessarily have to be in contiguous memory and not all of a process need be in memory at the same time.

- 45.** When is a logical address assigned to a variable?

When the program is compiled.

- 46.** When does address binding occur?

When the program is loaded into memory.

- 47.** How is memory divided in the single contiguous memory management approach?

Memory is divided into two sections, one for the operating system and one for the application program.

- 48.** When a program is compiled, where is it assumed that the program will be loaded into memory? That is, where are logical addresses assumed to begin?

At location 0.

- 49.** If, in a single contiguous memory management system, the program is loaded at address 30215, compute the physical addresses (in decimal) that correspond to the following logical addresses:

a. 9223

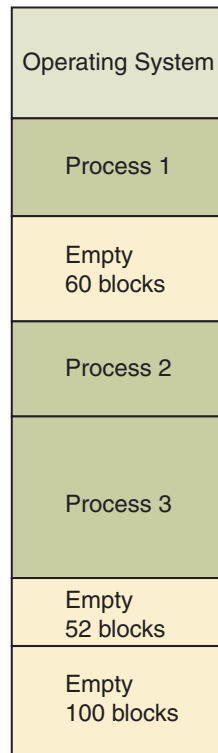
39438

b. 2302

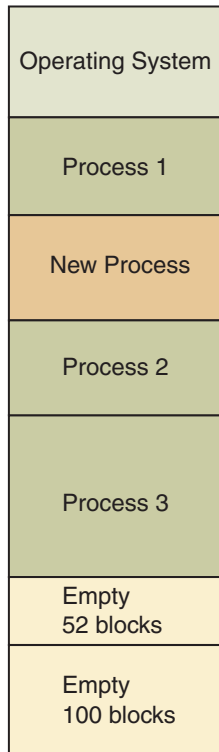
32517

- c. 7044
37259
50. In a single contiguous memory management approach, if the logical address of a variable is L and the beginning of the application program is A , what is the formula for binding the logical address to the physical address?
 $L + A$
51. If, in a fixed partition memory management system, the current value of the base register is 42993 and the current value of the bounds register is 2031, compute the physical addresses that correspond to the following logical addresses:
- a. 104
43097
- b. 1755
44748
- c. 3041
Address out of bounds of partition.
52. If more than one partition is being used (either fixed or dynamic), what does the bounds register contain?
The bounds register contains the length of the current partition.
53. If more than one partition is being used (either fixed or dynamic), what does the base register contain?
The base register contains the beginning address of the current partition.
54. Why is the logical address compared to the bounds register before a physical address is calculated?
The bounds register contains the length of the current partition. If the logical address is greater than the bounds register, then the physical address is not within the current partition.
55. If, in a dynamic partition memory management system, the current value of the base register is 42993 and the current value of the bounds register is 2031, compute the physical addresses that correspond to the following logical addresses:
- a. 104
43097
- b. 1755
44748
- c. 3041
Address out of bounds of partition.

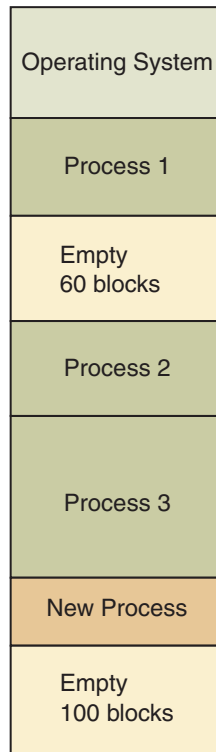
Exercises 56 and 57 use the following state of memory.

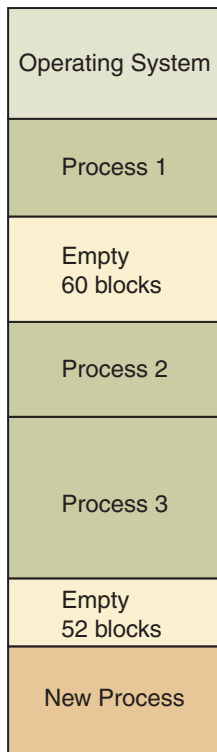


- 56.** If the partitions are fixed and a new job arrives requiring 52 blocks of main memory, show memory after using each of the following partition selection approaches:
- a.** first fit
 - b.** best fit
 - c.** worst fit

a. first fit

b. best fit

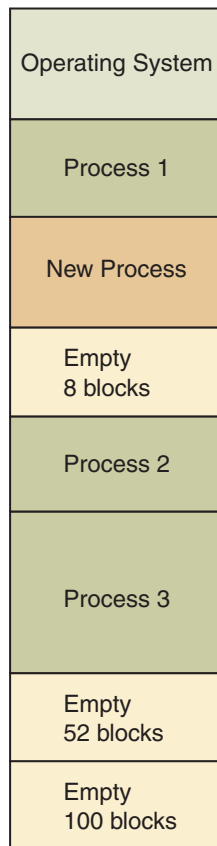


c. worst fit

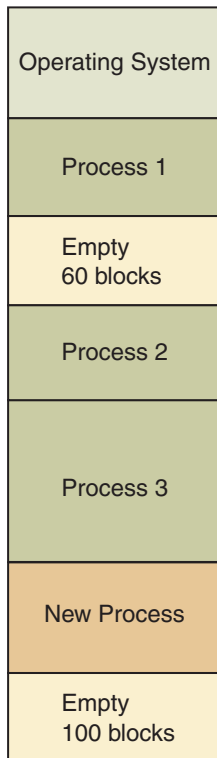
57. If the partitions are dynamic and a new job arrives requiring 52 blocks of main memory, show memory after using each of the following partition selection approaches:

- a.** first fit
- b.** best fit
- c.** worst fit

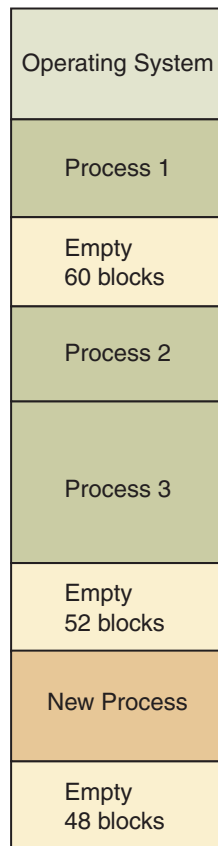
a. first fit



b. best fit



c. worst fit



- 58.** If a logical address in a paged memory management system is $\langle 2, 133 \rangle$, what do the values mean?

This address means the 133 byte on page 2.

Exercises 59–61 refer to the following PMT.

Page	0	1	2	3
Frame	5	2	7	3

- 59.** If the frame size is 1024, what is the physical address associated with the logical address $\langle 2, 85 \rangle$?

7253

- 60.** If the frame size is 1024, what is the physical address associated with the logical address <3,555>?

3627

- 61.** If the frame size is 1024, what is the physical address associated with the logical address <3,1555>?

Illegal address. The offset is larger than the page size.

- 62.** What is virtual memory and how does it apply to demand paging?

Virtual memory is the illusion that memory is limitless and thus there is no limit on the size of a program. Demand paging is the technique where pages are brought into memory only when they are referenced (needed). Demand paging allows programs of any size to run, thus giving the illusion of infinite memory.

- 63.** What are the conceptual stages through which a process moves while being managed by the operating system?

new, ready, running, waiting, and terminated

- 64.** Describe how a process might move through the various process states. Create specific reasons why this process moves from one state to another.

A new process begins in the new state. When the process has no bars to its execution, it moves into the ready state. It waits in the ready state until it gets time in the running state. It runs for a while and issues a command for file input. The process is moved into the waiting state until the I/O has been completed, at which time it moves into the ready state to await another turn in the running state. Eventually it gets back to the CPU and runs until it needs access to a part of the program that is on secondary storage. It moves into the waiting state until the needed pages are brought in; then it moves back to the ready state. It gets its third shot at the CPU and finishes, and moves into the terminated state.

- 65.** What is a process control block?

A process control block (PCB) is a data structure that contains information about a process. A PCB is created for each new process. When a process moves from one state to another, its PCB is moved with it.

- 67.** How is each conceptual stage represented in the OS?

Each conceptual stage is represented by a list of the PCBs in that stage.

- 68.** What is a context switch?

When a process is moved out of the CPU, the current contents of the registers, including the program counter, must be saved in the process's PCB. When a new process moves into the CPU, the contents

of the registers from this process's PCB are restored. This process of saving and restoring registers is called a context switch.

- 69.** Distinguish between preemptive scheduling and nonpreemptive scheduling.

With nonpreemptive scheduling, once a process is in the running state, it remains there until it voluntarily leaves. With preemptive scheduling, the OS can move a process from the running state to the waiting state or ready state.

- 70.** Name and describe three CPU scheduling algorithms.

First-come, first-served: The processes are moved into the running state in the order in which they arrive in the ready state.

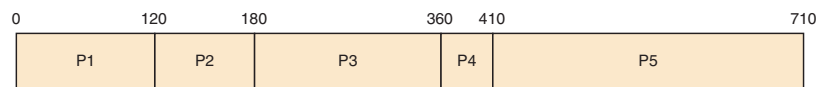
Shortest job next: When the CPU is ready for another job, the process in the ready state that takes the shortest time is moved into the running state. The estimated length of time that a process needs the CPU may or may not be accurate.

Round robin: Each process stays in the running state for a predetermined amount of time, called a time slice. When a process's time slice is over, it is moved back into the ready state, where it stays until it is its turn for the CPU again.

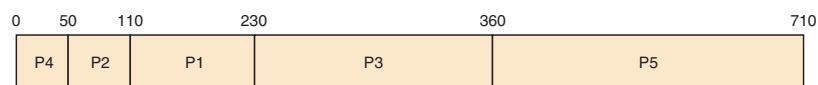
Use the following table of processes and service time for Exercises 71 through 73.

Process	P1	P2	P3	P4	P5
Service time	120	60	180	50	300

- 71.** Draw a Gantt chart that shows the completion times for each process using first-come, first-served CPU scheduling.



- 72.** Draw a Gantt chart that shows the completion times for each process using shortest job next CPU scheduling.

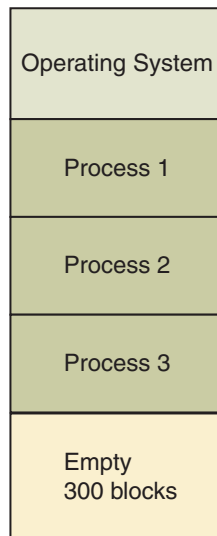


73. Draw a Gantt chart that shows the completion times for each process using round robin CPU scheduling with a time slice of 60.

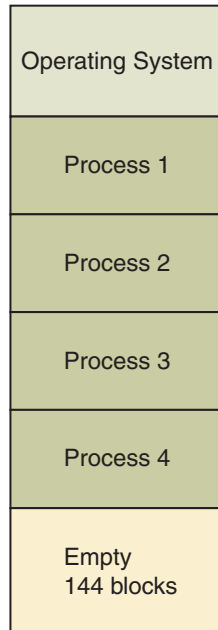
0	60	120	180	230	290	350	410	470	530	590	650	710
P1	P2	P3	P4	P5	P1	P3	P5	P3	P5	P5	P5	

74. Given the following state of memory where the partitions are dynamic, show memory after using each of the following partition selection approaches after a new job requiring 66 blocks of main memory.

- a. first fit
- b. best fit
- c. worst fit



The answer is the same in all three partition selection approaches.



- 75.** Distinguish between fixed partitions and dynamic partitions.

The sizes of the partitions are fixed in a fixed partition scheme, although they are not necessarily the same size. In a dynamic partition scheme, the partitions are allocated as needed.

Chapter 11 Exercises

For Exercises 1–15, mark the answers true and false as follows:

- A.** True
- B.** False

- 1.** A text file stores binary data that is organized into groups of 8 or 16 bits that are interpreted as characters.
A
- 2.** A program written in a high-level language is stored in a text file that is also called a source file.
A
- 3.** The type of a file determines what kinds of operations can be performed on it.
A

4. The current file pointer indicates the end of a file.
B
5. Sequential access and direct access take about the same amount of time to retrieve data.
B
6. Some operating systems maintain a separate read pointer and write pointer for a file.
A
7. Unix file permissions allow a group of users to access a file in various ways.
A
8. In most operating systems, a directory is represented as a file.
A
9. Two files in a directory system can have the same name if they are in different directories.
A
10. A relative path is relative to the root of the directory hierarchy.
B
11. An absolute path and a relative path will always be the same length.
B
12. An operating system is responsible for managing the access to a disk drive.
A
13. The seek time is the amount of time it takes for the heads of a disk to reach a particular cylinder.
A
14. The shortest-seek-time-first disk scheduling algorithm moves the heads the minimum amount it can to satisfy a pending request.
A
15. The first-come, first-served disk scheduling algorithm moves the heads the minimum amount it can to satisfy a pending request.
B

For Exercises 16–20, match the file extensions with the appropriate file.

- A.** txt
- B.** mp3, au, and wav
- C.** gif, tiff, jpg
- D.** doc and wp3
- E.** java, c, and cpp

16. audio file

B

17. image file

C

18. text data file

A

19. program source file

E

20. word processing file

D

For Exercises 21–23, match the symbol with its use.

- A.** /
- B.** \
- C.** ..

21. Symbol used to separate the names in a path in a Windows environment.

B

22. Symbol used to separate the names in a path in a Unix environment.

A

23. Symbol used to represent the parent directory in a relative path name.

C

Exercises 24–57 are problems or short answer questions.

24. What is a file?

A file is the smallest amount of information that can be written to secondary memory. It is a named collection of data, used for organizing secondary memory.

25. Distinguish between a file and a directory.

A file is a named collection of data. A directory is a named collection of files.

- 26.** Distinguish between a file and a file system.
A file is a named collection of data. A file system is the operating system's logical view of the files it manages.
- 27.** Why is a file a generic concept and not a technical one?
A file is just a named collection of bits (data) in storage. Because there are different operating systems, there are different technical views of a file. Because we are talking from the user's view, not the implementation view, the concept is generic.
- 28.** Name and describe the two basic classifications of files.
Text files: Files that contain text. Each byte is an ASCII character or each 2 bytes is a Unicode character.
Binary files: The bytes in a binary file do not necessarily contain characters. These files require a special interpretation.
- 29.** Why is the term binary file a misnomer?
All files ultimately are just a collection of bits, so why call one file type "binary?" In a binary file, the bits are not interpreted as text. A binary file would just be a stream of uninterpreted bits unless there is an interpretation provided. If a binary file is printed without interpretation, it looks like garbage.
- 30.** Distinguish between a file type and a file extension.
A file type is a description of the information contained in the file. A file extension is a part of the file name that follows a dot and identifies the file type.
- 31.** What would happen if you give the name "myFile.jpg" to a text file?
It depends on what application program you use to open the file. If you use a program that expects an image file, you would get an error. If you use a program that expects a text file, there would be no problem.
- 32.** How can an operating system make use of the file types that it recognizes?
If you click on a file on your desktop and the OS recognizes the file type, then the appropriate application program can be called to open the file. If you are writing Java programs using an integrated environment, then the files saved in the IDE are tagged, and clicking on a file automatically opens the file in the IDE.
- 33.** How does an operating system keep track of secondary memory?
The OS maintains a table indicating which blocks of memory are free. The OS also maintains a table for each directory that contains information about the files in that directory.

- 34.** What does it mean to open and close a file?

Operating systems keep a table of currently open files. The open operation enters the file into this table and places the file pointer at the beginning of the file. The close operation removes the file from the table of open files.

- 35.** What does it mean to truncate a file?

Truncating a file means that all the information on the file is erased but the administrative entries remain in the file tables. Occasionally, the truncate operation removes the information from the file pointer to the end.

- 36.** Compare and contrast sequential and direct file access.

Both sequential and direct file access find and access a record. In sequential access, the file pointer begins at the beginning of the file and can only move in one direction. Thus sequential access is linear: The only record that can be accessed is the first or the one immediately following the last one accessed. In direct access, the file pointer can be moved to any specific record and the data accessed from that place.

- 37.** File access is independent of any physical medium.

- a.** How could you implement sequential access on a disk?

Sequential access always accesses the next record. You implement sequential access on a disk by not giving the user an access command that takes a record address as a parameter.

- b.** How could you implement direct access on a magnetic tape?

Each record on a magnetic tape is conceptually numbered from the first to the last. Keep a counter of which record was read last. When a user gives an access command to read a specific record, if the record number is beyond the last record read, then records are read and skipped until the correct record is found. If the record number comes before the last record read, the tape is rewound and records are read and skipped until the correct record is found.

- 38.** What is a file protection mechanism?

A file protection mechanism is one that an operating system implements that ensures that only valid users can access a particular file.

- 39.** How does Unix implement file protection?

Unix implements file protection by associating with each file a 3x3 table in which the rows are Owner, Group, and World, and the columns are Read, Write/Delete, and Execute. The contents of each cell in the table are boolean values meaning yes and no. For example, a yes in the cell (Owner, Execute) means that the owner of the file can

execute it. A no in the cell (World, Write/Delete) means that permission to write or delete a file is not granted to anyone that is not the owner of the file or within a specified group. (Group is a list of those considered part of the group.)

- 40.** Given the following file permission, answer these questions.

	Read	Write/Delete	Execute
Owner	Yes	Yes	Yes
Group	Yes	Yes	No
World	Yes	No	No

- a.** Who can read the file?
Anyone can read the file.
 - b.** Who can write or delete the file?
The owner and members of the group can write or delete the file.
 - c.** Who can execute the file?
Only the owner can execute the file.
 - d.** What do you know about the content of the file?
Because the owner has permission to execute the file, it must contain an executable program.
- 41.** What is the minimum amount of information a directory must contain about each file?
A directory must contain the file name, the file type, the address on disk where the file is stored, the current size of the file, and permission information.
- 42.** How do most operating systems represent a directory?
As a file.
- 43.** Answer the following questions about directories.
- a.** A directory that contains another directory is called what?
parent directory
 - b.** A directory contained within another directory is called what?
subdirectory
 - c.** The directory that is not contained in any other directory is called what?
root directory

- d.** The structure showing the nested directory organization is called what?

directory tree

- e.** Relate the structure in d to the binary tree data structure examined in Chapter 9.

A directory tree and a binary tree are both hierarchical structures in which there is only one way to reach any subtree. The root directory is equivalent to the root of the binary tree. In a binary tree, each node can have none, one, or two child nodes. In a directory tree, each node can have any number of subdirectories.

- 44.** What is the directory called in which you are working at any one moment?

working directory

- 45.** What is a path?

A path is a text string that specifies the location of a file or subdirectory.

- 46.** Distinguish between an absolute path and a relative path.

An absolute path is a path that begins at the root directory and includes all successive subdirectories. A relative path is a path that begins at the current working directory and includes all successive subdirectories.

- 47.** Show the absolute path to each of the following files or directories using the directory tree shown in Figure 11.4:

- a.** QTEffects.qtx

C:\WINDOWS\System\QuickTime\QTEffects.qtx

- b.** brooks.mp3

C:\My Documents\downloads\brooks.mp3

- c.** Program Files

C:\Program Files

- d.** 3dMaze.scr

C:\WINDOWS\System\3dMaze.scr

- e.** Powerpnt.exe

C:\Program Files\MS Office\Powerpnt.exe

- 48.** Show the absolute path to each of the following files or directories using the directory tree shown in Figure 11.5:

- a.** tar

/bin/tar

- b.** access.old

/etc/mail/access.old

- c. `named.conf`
`/etc/named.conf`
 - d. `smith`
`/home/smith`
 - e. `week3.txt`
`/home/smith/reports/week1.txt`
 - f. `printall`
`/home/jones/utilities/printall`
- 49.** Assuming the current working directory is `C:\WINDOWS\System`, give the relative path name to the following files or directories using the directory tree shown in Figure 11.4:
- a. `QTImage.qtx`
`QuickTime\QTImage.qtx`
 - b. `calc.exe`
`..\calc.exe`
 - c. `letters`
`...\My Documents\letters`
 - d. `proj3.java`
`...\My Documents\csc101\proj3.java`
 - e. `adobep4.hlp`
`adobep4.hlp`
 - f. `WinWord.exe`
`...\Program Files\MS Office\Winword.exe`
- 50.** Show the relative path to each of the following files or directories using the directory tree shown in Figure 11.5.
- a. `localtime` when working directory is the root directory
`/etc/localtime`
 - b. `localtime` when the working directory is `etc`
`localtime`
 - c. `printall` when the working directory is `utilities`
`printall`
 - d. `week1.txt` when the working directory is `man2`
`../reports/week1.txt`
- 51.** What is the worst bottleneck in a computer system?
Transferring data to and from secondary memory is the worst bottleneck.

- 52.** Why is disk scheduling concerned more with cylinders than with tracks and sectors?

Seek time (the time to find the right cylinder) is more time consuming than locating which track or which sector, so seek time is the time to minimize.

- 53.** Name and describe three disk scheduling algorithms.

First-come, first-serve (FCFS): The requests are handled in the order in which they are generated.

Shortest-seek-time-first (SSTF): The request closest to the read/write heads is handled next.

SCAN: The read/write heads move back and forth handling the closest in the direction in which they are moving.

Use the following list of cylinder requests in Exercises 54 through 57. They are listed in the order in which they were received.

40, 12, 22, 66, 67, 33, 80

- 54.** List the order in which these requests are handled if the FCFS algorithm is used. Assume that the disk is positioned at cylinder 50.

40, 12, 22, 66, 67, 33, 80

- 55.** List the order in which these requests are handled if the SSTF algorithm is used. Assume that the disk is positioned at cylinder 50.

40, 33, 22, 12, 66, 67, 80

- 56.** List the order in which these requests are handled if the SCAN algorithm is used. Assume that the disk is positioned at cylinder 50 and the read/write heads are moving toward the higher cylinder numbers.

66, 67, 80, 40, 33, 22, 12

- 57.** Explain the concept of starvation.

In the SSTF algorithm, it is possible for some requests never to be serviced because requests closer to the read/write heads keep being issued.

Chapter 12 Exercises

For Exercises 1–18, mark the answers true and false as follows:

- A.** True
- B.** False

- 1.** A cell in a spreadsheet can contain only raw data.

B

- 2.** The values in a spreadsheet can be formatted in a variety of ways.

A

3. A spreadsheet should be set up so that changes to the data are automatically reflected in any cells affected by that data.
A
4. A spreadsheet function is a program that the user writes to compute a value.
B (they are built into the software)
5. A range of cells can be specified that go horizontally or vertically, but not both.
B
6. A circular reference in a spreadsheet is a powerful and useful situation.
B
7. A spreadsheet is useful for performing what-if analysis.
A
8. What-if analysis can only affect one value at a time in a spreadsheet.
B
9. A database engine is software that supports access to the database contents.
A
10. The physical database represents the logical structure of the data in the database.
B
11. A query is a request to a database for information.
A
12. The results of a query can be structured in many ways.
A
13. The hierarchical model is the most popular database management model today.
B
14. A database table is a collection of records, and a record is a collection of fields.
A
15. The values in the key fields of a table uniquely identify a record among all other records in the table.
A
16. A database engine often interacts with a particular language for accessing and modifying the database.
A

- 17.** An entity-relationship (ER) diagram represents primary database elements in a graphical form.

A

- 18.** The cardinality of a relationship puts restrictions on the number of relationships that can exist at one time.

A

For Exercises 19–23, match the solution to the question using the spreadsheet shown.

- A. dynamic
- B. function
- C. circular
- D. range
- E. schema
- F. field

- 19.** A spreadsheet is ____ in that it responds to changes in the data by immediately updating all affected values.

A

- 20.** A spreadsheet formula may operate on a ____ of cells, such as C4..C18.

D

- 21.** The database ____ is the specification of the logical structure of the data in the database.

E

- 22.** A ____ reference occurs when the result of one formula is ultimately based on another, and vice versa.

C

- 23.** A ____ contains a single data value.

F

Exercises 24–62 are problems or short answer questions.

Use the following spreadsheet that contains student grades for Exercises 24–32.

	A	B	C	D	E	F	G	H
1					Grades			
2				Exam 1	Exam 2	Exam 3	Average	
3								
4			Bill	89	33	80	67.3333	
5			Bob	90	50	75	71.6666	
6			Chris	66	60	70	65.3333	
7			Jim	50	75	77	67.3333	
8		Students	Judy	80	80	80	80	
9			June	83	84	85	84	
10			Mari	87	89	90	88.6666	
11			Mary	99	98	90	95.6666	
12			Phil	89	90	85	88	
13			Sarah	75	90	85	83.3333	
14			Suzy	86	90	95	90	
15		Total		893	839	912	881.333	
16		Average		81.1818	76.2727	82.9090	80.1212	

24. Specify the grades for Exam 2.

E4..E14

25. Specify the average for Exam 1.

D15

26. Specify the average for Sarah.

G13

27. Specify the third exam grade for Mari.

F10

28. Specify the exam grades for Susy.

D14..F14

29. What formula is stored in F15?

SUM(F4..F14)

- 30.** D16 contains the formula `D15/COUNT(D4..D14)`. What is another formula for the same value?

`AVERAGE(D4..D14)`

- 31.** What formula is stored in E13?

There isn't a formula stored there; it is a single value.

- 32.** What values in what cells would change if Phil's Exam 2 score was corrected to 87?

F12, F15, F16, G12, G15

- 33.** What is a spreadsheet circular reference? Why is it a problem?

A circular reference is one in which two or more formulas depend upon each other. For example, F5 may use D4 in its calculation, and D4 may use F5 in its calculation. A circular reference may be direct as in the F5, D4 example or indirect involving many different formulas. The problem is that the calculation cannot be made because of the interdependence of the references.

- 34.** Give a specific example of an indirect circular reference similar to the one shown in Figure 12.5.

`B1 = SUM(A1..A5) * C2`

`C2 = SUM(B5..B12)`

`B8 = G1 - D2`

`D2 = G3 * B1`

B1 depends on C2; C2 depends on B8; B8 depends on D2; and D2 depends on B1.

- 35.** What is what-if analysis?

The values representing assumptions in a spreadsheet can be changed and the effects on related data can be observed. This varying of values and the observation of the results represent what-if analysis. That is, you can ask what happens if you change certain values and see the results.

- 36.** Name some what-if analysis questions that you might ask if you were using a spreadsheet to plan and track some stock purchases. Explain how you might set up a spreadsheet to help answer those questions.

We are considering buying stock in four Fortune Five Hundred companies.

What would my gain be if all four of the stocks gained 6% over a year?

What would my gain be if two gained 6% over a year, but the other two lost 2%?

What would my loss be if all four lost 2%?

All of these percentages could be varied and the results examined.

The spreadsheet could be set up with the stock names in B2..B5, the initial prices in C2..C5, the calculated prices based on percentage gains or losses per stock in D2..D5. C6 could contain the formula for summing C2..C5. D6 could contain the formula for summing D2..D5. E6 could contain the formula for subtracting C6 from D6 to calculate the gain or loss.

For Questions 37 through 40, use the paper spreadsheet form supplied on the textbook's website or use an actual spreadsheet application program to design the spreadsheets. Your instructor may provide more specific instructions regarding these questions.

These are activities for which there are no specific answers.

- 37.** Design a spreadsheet to track the statistics of your favorite major league baseball team. Include data regarding runs, hits, errors, and runs-batted-in (RBIs). Compute appropriate statistics for individual players and the team as a whole.
- 38.** Design a spreadsheet to maintain a grade sheet for a set of students. Include tests and projects, giving various weights to each in the calculation of the final grade for each student. Compute the average grade per test and project for the whole class.
- 39.** Assume you are going on a business trip. Design a spreadsheet to keep track of your expenses and create a summary of your totals. Include various aspects of travel such as car mileage, flight costs, hotel costs, and miscellaneous (such as taxis and tips).
- 40.** Design a spreadsheet to estimate and then keep track of a particular project's activities. List the activities, the estimated and actual dates for those activities, and schedule slippage or gain. Add other data as appropriate for your project.
- 41.** Compare a database with a database management system.

A database is a structured set of data. A database management system is a software system made up of the database, a database engine (for manipulating the database), and a database schema that provides the logical view of the database.

- 42.** What is a database schema?

A database schema is the specification of the logical structure of a database.

- 43.** Describe the general organization of a relational database.

A relational database is organized around the concept of a table. Both data and relationships are represented in tables.

- 44.** What is a field (attribute) in a database?

A field is a single value in a database record. A field represents one aspect or attribute of the item in a database.

- 45.** What other fields (attributes) might we include in the database table of Figure 12.7?

There are four fields in the table in Figure 12.7 that describe a movie: MovieID, Title, Genre, and Rating. Additional items might be Director, MaleLead, FemaleLead, Producer, and Date.

- 46.** What other fields (attributes) might we include in the database table of Figure 12.8?

There are four fields in the table in Figure 12.8 that describe a customer: CustomerID, Name, Address, and CreditCardNumber. Additional items might be PhoneNumber, HighestRatingAcceptable, and MaxMovies.

- 47.** What is a key in a relational database table?

In a relational database, every record in a table must be uniquely identified. The field or combination of fields that are used to uniquely identify records in a table is called the key field.

- 48.** Specify the schema for the database table of Figure 12.8.

Customer(CustomerId, Name, Address, CreditCardNumber)

- 49.** How are relationships represented in a relational database?

In a table, of course! A table is created that represents the relationship. Usually, the keys of both items in the relationships are represented as fields in the relationship table, along with appropriate information about the relationship.

- 50.** Define an SQL query that returns all attributes of all records in the Customer table.

```
select * from Customer
```

- 51.** Define an SQL query that returns the movie id number and title of all movies that have an R rating.

```
select MovieId, Title from Movie where Rating = 'R'
```


- 52.** Define an SQL query that returns the address of every customer in the Customer table that lives on Lois Lane.

```
select Address from Customer where Address like  
'%Lois Lane%'
```

- 53.** Define an SQL statement that inserts the movie Armageddon into the Movie table.

```
insert into Movie values(1433, Armageddon, action sci-  
fi, R)
```

- 54.** Define an SQL statement that changes the address of Amy Stevens in the Customer table.

```
update Customer set Address = '333 Silver Way' where  
Name = 'Amy Stevens'
```

- 55.** Define an SQL statement that deletes the customer with a customer id of 103.

```
delete from Customer where CustomerId = 103
```

- 56.** What is an ER diagram?

An ER diagram is a graphical representation of an entity-relationship model, which is a technique for designing relational databases.

- 57.** How are entities and relationships represented in an ER diagram?

In an ER diagram, tables are represented by rectangles, and diamonds represent relationships.

- 58.** How are attributes represented in an ER diagram?

Attributes are represented in ovals with lines connecting them to the entity they describe.

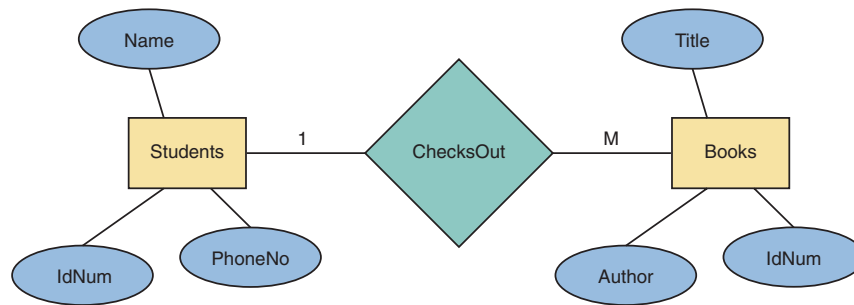
- 59.** What are cardinality constraints and how are they shown in ER diagrams?

Cardinality constraints are the number of relationships that can exist between entities in an ER diagram. They are represented by the number 1 and the letter M (for Many). These values are written on the line connecting the table and the relationship.

- 60.** What are the three general cardinality constraints?

One to one, one to many, and many to many.

61. Design a database that stores data about the books in a library, the students that use them, and the ability to check out books for a period of time. Create an ER diagram and sample tables.

**Students**

IdNum	Name	PhoneNo
223456789	Sally Silent	232-4432
432543654	Lottie Loud	343-3321
898978675	Maud Middle	454-3452

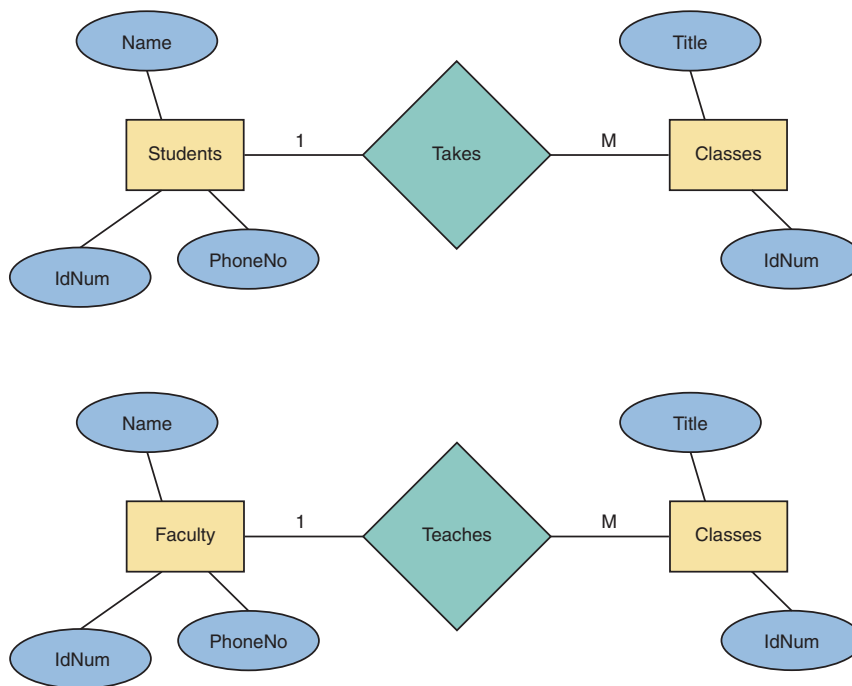
Books

IdNum	Author	Title
443233	Brown	Fly Fishing in the Sahara
567622	Smith	Dust Storms in the Amazon Basin
657687	Anderson	Skating on Thin Ice

ChecksOut

IdNum	Author	DateDue
443233	Brown	10/23/2004
567622	Smith	10/30/2004
657687	Anderson	11/1/2004

62. Design a database that stores data about the courses taught at a university, the professors that teach those courses, and the students that take those courses. Create an ER diagram and sample tables.



Students

IdNum	Name	PhoneNo
223456789	Sally Silent	232-4432
432543654	Lottie Loud	343-3321
898978675	Maud Middle	454-3452

Faculty

IdNum	Name	PhoneNo
344567777	Abby Absent	632-4432
222222222	Steven Stern	373-3321
434343434	Mary Modern	453-3452

Courses

IdNum	Title
CS304	Introduction to CS
CS328	Abstract Data Types
CS375	Compiler Design

Chapter 13 Exercises

For Exercises 1–5, match the type of ambiguity with an example.

- A.** Lexical
- B.** Referential
- C.** Syntactic

1. “Stand up for your flag.”
 - A. Does “stand up” mean support or rise to your feet?
2. “Go down the street on the left.”
 - C. Does this mean to go left down the street or go down the street that is on the left?
3. “He drove the car over the lawn mower, but it wasn’t hurt.”
 - B. What wasn’t hurt, the car or the lawn mower?

4. "I saw the movie flying to Houston."
C
5. "Mary and Kay were playing until she came inside."
B

For Exercises 6–21, mark the answers true and false as follows:

- A. True
B. False
6. A computer does some tasks much better than a human being.
A
7. A human being does some tasks much better than a computer.
A
8. A computer system that can pass the Turing test is considered to be intelligent.
A
9. Some AI researchers don't think we can achieve true artificial intelligence until a computer processes information in the same way the human mind does.
A
10. A semantic network is used to model relationships.
A
11. If information is stored in a semantic network, it is easy to answer questions about it.
B (it depends on how the network is structured)
12. A computer has never beaten a human at chess in master-level play.
B
13. An inference engine is part of a rule-based expert system.
A
14. A biological neuron accepts a single input signal and produces multiple output signals.
B
15. Each element in an artificial neural net is affected by a numeric weight.
A
16. Voice synthesis is the most difficult part of natural language processing.
B

- 17.** Each human has a unique voiceprint that can be used to train voice recognition systems.

A

- 18.** The word “light” can be interpreted in many ways by a computer.

A

- 19.** Syntactic ambiguity is no longer a problem for natural language comprehension.

B

- 20.** A robot may follow the Sense-Plan-Act approach to control its movements.

A

- 21.** Isaac Asimov created three fundamental laws of robotics.

A

For Exercises 22–30, match the task with which (human or computer) can solve it more easily.

A. Computer

B. Human

- 22.** Identify a dog in a picture.

B

- 23.** Add a column of 100 four-digit numbers.

A

- 24.** Interpret a poem.

B

- 25.** Match a fingerprint.

A

- 26.** Paint a landscape.

B

- 27.** Carry on an intelligent conversation.

B

- 28.** Learn to speak.

B

- 29.** Judge guilt or innocence.

A

- 30.** Give affection.

A

Exercises 31–76 are problems or short answer questions.

31. What is the Turing Test?

The Turing Test is a test devised by Alan Turing to answer the question “How can we know we’ve succeeded in creating a machine that can think?” The test is based on whether a computer could fool a human into believing that the computer is another human being.

32. How is the Turing Test organized and administered?

A human interrogator sits in a room and uses a computer terminal to communicate with two respondents. The interrogator knows that one respondent is human and the other is a computer. After conversing with both the human and the computer, the interrogator must decide which respondent is the computer. If the computer could fool enough interrogators, then it must be considered intelligent.

33. What is weak equivalence and how does it apply to the Turing Test?

Weak equivalence is the equality of two systems based on their results. The Turing Test shows weak equivalence.

34. What is strong equivalence?

Strong equivalence is the quality of two systems based on their results and the process by which they arrive at those results.

35. What is the Loebner Prize?

Loebner Prize is the first formal instantiation of the Turing Test. It has been held annually since 1991.

36. Name and briefly describe five issues in the world of AI covered in this chapter.

Knowledge representation: The techniques used to represent knowledge so that a computer system can use it in problem solving.

Expert systems: Computer systems that embody the knowledge of human experts.

Neural networks: Computer systems that mimic the processing of the human brain.

Natural language processing: Computer systems that process the language that humans use to communicate.

Robotics: The study of mobile robots that use AI techniques to interact with their environments.

37. Name and define two knowledge representation techniques.

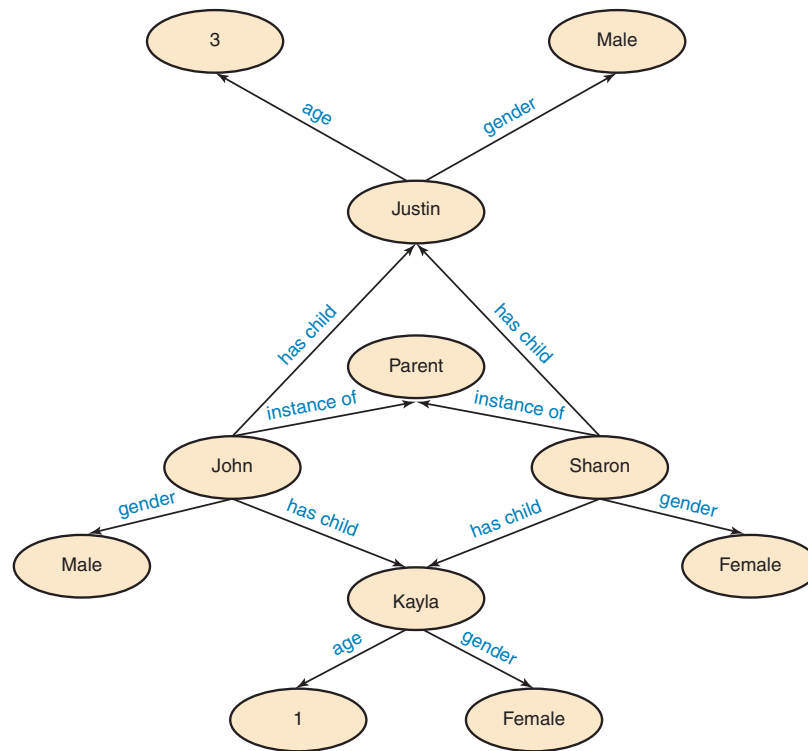
Semantic networks: A technique that represents the relationships among objects.

Search trees: A structure that represents alternatives in adversarial situations such as games.

- 38.** What data structure defined in Chapter 9 is used to represent a semantic network?

A graph is used to represent a semantic network. The nodes in the graph represent objects and the arrows (arcs) represent relationships.

- 39.** Create a semantic network for the relationships among your family members. List five questions that your semantic net could easily be used to answer and five questions that would be more of a challenge to answer.



Easy questions to answer given this organization:

Who are John's children?

What is the gender of Kayla?

How old are Sharon's children?

How many female children does John have?

Does Sharon have any children older than 5 years of age?

More challenging questions to answer given this organization:

Who are Kayla's parents?

Who are Justin's siblings?

How many female children are there?

Who is the mother of John's children?

Does John have any step-children?

- 40.** Create a semantic network that captures the information in a small section of a newspaper article.

This is an activity for which no answer is appropriate.

- 41.** What object-oriented properties do semantic networks borrow?

Semantic networks borrow inheritance and instantiation. The inheritance is expressed in the “is-a” relationship, and instantiation is expressed when an object is related to something that describes it.

- 42.** What is a search tree?

A search tree is a structure that represents all possible moves for both players in a two-person game.

- 43.** Why are trees for complex games like chess so large?

A search tree contains all possible moves from the first position, all possible moves from each of the moves from the first position, ..., all possible moves from all possible moves at the level above. Thus the trees are very large for complex games like chess.

- 44.** Distinguish between depth-first searching and breadth-first searching.

Depth-first searching begins at the top level (root) and continues going deeper and deeper into the tree until the search has reached a leaf node, at which time the search moves back up one level and starts down again. A breadth-first search begins at the top level, then searches every node on the next lower level, then searches every node at the next lower level, until it has searched every node on every level.

- 45.** What does it mean to prune a tree?

Pruning a tree means to eliminate some branches from searching.

- 46.** Distinguish between knowledge-based systems and expert systems.

A knowledge-based system is a software system that uses a specific set of information from which it extracts and processes particular pieces. An expert system is sometimes used as a synonym, but it can also

carry with it the idea of modeling the expertise of a professional in that particular field.

- 47.** Distinguish between rule-based systems and inference engines.

A rule-based system is a software system that uses a set of rules to guide its processing. An inference engine is the software system that processes the rules.

- 48.** What is an example of a human expert system?

A doctor is an example of a human expert system. The doctor asks questions and runs tests based on his knowledge and experience.

- 49.** What do we call a knowledge-based system that models the expertise of professionals in the field?

An expert system.

- 50.** Why is an expert system called a rule-based system?

An expert system is called a rule-based system because it uses a set of rules to guide its processing.

- 51.** What is the part of the software in an expert system that determines how the rules are followed and what conclusions can be drawn?

An inference engine.

- 52.** How are the rules expressed in an expert system?

The rules are expressed as selection statements (*if* statements).

- 53.** What are the advantages of an expert system?

An expert system is goal-oriented; it doesn't focus on abstract or theoretical information. It is efficient; it records previous responses and doesn't ask irrelevant questions. An expert system can provide useful guidance even if it can't provide the answer to a specific question.

- 54.** What is a single cell that conducts a chemically-based electronic signal?

A neuron

- 55.** What do a series of connected neurons form?

A pathway in the brain

- 56.** Upon what does the signal along a particular pathway depend?

The signals depend on the state of the neurons through which the signal passes.

- 57.** What are the multiple input tentacles in a biological neuron?

Dendrites

- 58.** What is the primary output tentacle in a biological neuron?

An axon

- 59.** From where do dendrites of one neuron pick up the signals from other neurons to form a network?

The dendrites of one neuron pick up the signals from the axons of other neurons to form a neural network.

- 60.** What is the gap between an axon and a dendrite?

A synapse.

- 61.** What tempers the strength of a synapse?

The chemical composition of a synapse tempers the strength of its input signal.

- 62.** What is the role of a synapse?

The role of a synapse is to weigh the input signal.

- 63.** How is a synapse modeled in an artificial neural network?

A synapse is represented by a weight assigned to each input signal.

- 64.** What is an effective weight in an artificial neuron?

An effective weight is the sum of the weights multiplied by the corresponding input values.

- 65.** How is the output value from an artificial neuron calculated?

Each neuron has a numeric threshold value. If the effective weight is greater than the threshold, a 1 is output; otherwise, a 0 is output.

- 66.** If the processing element in an artificial neural net accepted five input signals with values of 0, 0, 1, 1, and 0 and corresponding weights of 5, -2, 3, 3, and 6, what is the output if the threshold is 5?

1

- 67.** If the processing element in an artificial neural net accepted five input signals with values of 0, 0, 1, 1, and 0 and corresponding weights of 5, -2, 3, 3, and 6, what is the output if the threshold is 7?

0

- 68.** What is a phoneme?

A phoneme is a fundamental sound in a language.

- 69.** Describe the two distinct ways that voice synthesis can be accomplished.

In dynamic voice generation, the set of phonemes for a language are generated. A computer examines the letters that make up a word and produce the sequence of sounds using the language's phonemes.

In recorded speech, human speech is recorded. A computer chooses the correct word from its file of recorded words.

Dynamic voice generation can make an attempt to pronounce any word, but recorded speech can only pronounce words that have been prerecorded.

- 70.** What issues affect the ability to recognize the words spoken by a human voice?

Accents, regional dialects, voice pitch, homonyms, and the clarity of a person's speech.

- 71.** How can a voice recognition system be trained?

A voiceprint is a plot of frequency changes over time representing the sound of a human's speech. To train a voice recognition system, a person says the same word several times and the computer records an average voiceprint for the word.

- 72.** Why are personalized voice recognition systems so much better than those that are not specific to one person?

Generalized systems have to use generic voiceprints, but personalized systems can use voiceprints specific to the user.

- 73.** Name and describe two categories of robots.

Fixed robots: Robots that remain in one place to accomplish their task.

Mobile robots: Robots that move around, thus having to interact with their environment.

- 74.** What are planning systems?

Planning systems are large software systems that, given a goal, a starting position, and an ending situation, generate an algorithm for a solution.

- 75.** What defines subsumption architecture?

Behaviors run in parallel unless they come into conflict, at which time the ordering of goals determine which behavior takes precedence.

- 76.** Of what is a robot composed?

A robot is composed of sensors, actuators, and computational elements. The sensors take in data about the outside world, the actuators move the robot, and the computational elements send instructions to the actuators.

Chapter 14 Exercises

For Exercises 1–8, match the kind of simulation with the example.

- A.** Continuous simulation
- B.** Discrete event simulation

1. Weather forecasting

A

2. Stock portfolio modeling

B

3. Seismic exploration

A

4. Hurricane tracking

A

5. Predicting number of tellers a new bank needs

B

6. Determining the number of waiting rooms necessary for a doctor's office

B

7. Gas exploration

A

8. Air-chemistry models

A

For Exercises 9–20, mark the answers true and false as follows:

- A.** True
- B.** False

9. Simple systems are best suited to being simulated.

B

10. Complex systems are dynamic, interactive, and complicated.

A

11. A model is an abstraction of a real system.

A

12. The representation of a model may be concrete or abstract.

A

13. In computer simulations, the models are concrete.

B

- 14.** The more characteristics or features represented in the model the better.
B
- 15.** Continuous simulations are represented by entities, attributes, and events.
B
- 16.** Changes in discrete event simulations are represented by partial differential equations.
B
- 17.** CAD stands for computer-aided drafting.
B
- 18.** A time-driven simulation can be thought of as a big loop that executes a set of rules for each value of the clock.
A
- 19.** A model whose realization is within a computer program is an abstract model.
A
- 20.** A concrete model can be realized within a computer program.
B

Exercises 21–49 are problems or short answer questions.

- 21.** Define simulation and give five examples from everyday life.
Simulation is the development of a model of a complex system and the experimentation with the model to observe the results. TV weather forecasters use computer models to predict the weather. Pilots spend time in a flight simulator before actually flying an aircraft. Engineers use wind tunnels to test out a new design. Automotive engineers use simulated crash tests to see how cars survive at different speeds. Dummies are used in crash tests to see how they survive. A chef experiments with a new recipe to see which combination of ingredients is best.
- 22.** What is the essence of constructing a model?
The essence of constructing a model is to identify a small subset of characteristics or features that are sufficient to describe the behavior to be investigated.
- 23.** Name two types of simulations and distinguish between them.
Continuous simulation treats time as continuous and expresses changes in terms of a set of differential equations that reflect the rela-

tionships among the set of characteristics. Discrete event simulation is made up of entities, attributes, and events, where entities represent objects in the real system, attributes are characteristics of a particular entity, and events are interactions among entities.

- 24.** What are the keys to constructing a good model?

The keys to constructing a good model are correctly choosing the entities to represent the system and correctly determining the rules that define the results of the events.

- 25.** What defines the interactions among entities in a discrete event simulation?

A set of rules that are part of the model determine the interactions among the events.

- 26.** What is the relationship between object-oriented design and model building?

Abstract models are implemented in a computer program, so object-oriented design techniques can be used to build the model.

- 27.** Define the goal of a queuing system.

The goal of a queuing system is to determine how to minimize wait time.

- 28.** What are the four necessary pieces of information needed to build a queuing system?

The four necessary steps are

- the number of events and how they affect the system (to determine the rules of entity interaction)
- the number of servers (entities)
- the distribution of arrival times (to determine if an entity enters the system)
- the expected service time (to determine the duration of an event)

- 29.** What part does a random number generator play in queuing simulations?

The random number generator is used to represent luck. If an event happens every x minutes, a random number generator is used to determine if the event happens at each minute.

- 30.** Write the rules for a queuing simulation of a one-pump gas station, where a car arrives every 3 minutes and the service time is 4 minutes.

If a car arrives, it gets in line. A car arrives if the random number is between 0.0 and 0.33.

If the pump is free and there is a car waiting, the first car in line leaves the line and goes to the pump and the service time is set to 4.

If a car is at the pump, the time remaining for the car is decremented.

If there are cars in line, the additional minutes that they have been waiting are recorded.

- 31.** Do you think the gas station in Exercise 30 will be in business very long? Explain.

No. The service time is greater than the arrival probability.

- 32.** Rewrite the simulation in Exercise 30 such that a car arrives every 2 minutes and the service time is 2 minutes.

If a car arrives, it gets in line. A car arrives if the random number is between 0.0 and 0.5.

If the pump is free and there is a car waiting, the first car in line leaves the line and goes to the pump and the service time is set to 2.

If a car is at the pump, the time remaining for the car is decremented.

If there are cars in line, the additional minutes that they have been waiting are recorded.

- 33.** Write the rules for a queuing system for an airline reservation counter. There is one queue and two reservation clerks. People arrive every 3 minutes and take 3 minutes to be processed.

If a customer arrives, he or she gets in line. A customer arrives if the random number is between 0.0 and 0.33.

If a clerk is free and there is a person waiting, the first person in line leaves the line and goes to the free clerk and the service time is set to 3.

If a customer is with the clerk, the time remaining for the customer is decremented.

If there are customers in line, the additional minutes that they have been waiting are recorded.

- 34.** Distinguish between a FIFO queue and a priority queue.

Deque in a FIFO queue returns the entity that has been in the queue the longest time. Deque in a priority queue returns the entity with the highest priority.

- 35.** What did SIMULA contribute to object-oriented programming methodology?

SIMULA introduced the concepts of classes and objects, inheritance, and polymorphism.

- 36.** In general, meteorological models are based on the time-dependent equations of what fields?

Meteorological models are based on time-dependent equations from fluid mechanics and thermodynamics.

- 37.** How much mathematics is necessary to be a meteorologist?

Five college courses in the calculus sequence plus a course or two in numerical methods should provide the background to understand the mathematics involved in these models.

- 38.** Why is there more than one weather prediction model?

Different models exist because different assumptions are possible.

- 39.** Why do different meteorologists give different forecasts if they are using the same models?

Meteorologists may or may not agree with the predictions from a particular model. Also, various models give conflicting information. Thus the meteorologist must use their judgment as to which, if any, model is correct.

- 40.** What are specialized meteorological models and how are they used?

Specialized meteorological models are adaptations for specialized research purposes. A meteorological model may be combined with air-chemistry models to diagnose atmospheric transport and diffusion for a variety of air-quality applications. Specialized meteorological models are useful in the military and aviation industries.

- 41.** What are seismic models used for?

Seismic models depict the propagation of seismic waves through the earth's medium. They are used for oil and mineral exploration.

- 42.** What are 2-dimensional CAD models used for?

Two-dimensional CAD models are used as electronic drawing boards.

- 43.** What are 3-dimensional CAD models used for?

Three-dimensional CAD models are used for geometric modeling; that is, three-dimensional objects. They can be used for modeling anything from cars to houses.

- 44.** What are the 3 methods of modeling in three dimensions and how do you determine which method should be used where?

The three types of three-dimensional models are wireframe, in which objects are represented by line elements; surface modeling, in which the outside of an object is modeled; and solid modeling, in which both the interior and exterior of an object are modeled. The intended purpose of the image dictates the appropriate model.

- 45.** Distinguish between an embedded system and a regular computing system.

Embedded systems are computers that are dedicated to perform a narrow range of functions as part of a larger system. There is only minimal end-user or operator intervention, if any at all.

- 46.** Embedded systems' programmers are the last holdout for assembly language programming. Explain.

In embedded systems, the size of the code and the speed of execution are very important. Assembly-language programs provide the best opportunity for a programmer to streamline and speed up the code.

- 47.** A random number generator can be used to vary service times as well as determine arrivals. For example, assume that 20% of the customers take 8 minutes and 80% of the customers take 3 minutes. How might you use a random number generator to reflect this distribution?

Generate a random number for each customer. If the number is between 0 and .20, the customer takes 8 minutes; otherwise the customer takes 3 minutes.

- 48.** Why do we say that simulation doesn't give an answer?

Simulation is a tool that allows you to investigate "what if" questions. You can try different values for the parameters of the simulation and see what happens to the wait time.

- 49.** What do simulations and spreadsheet programs have in common?

Simulations and spreadsheet programs both examine "what if" questions.

Chapter 15 Exercises

For Exercises 1–6, match the word or acronym with the definition or the appropriate blank.

- A.** LAN
- B.** WAN
- C.** Gateway
- D.** Bus topology
- E.** Ethernet
- F.** Internet

- 1.** The Internet is a _____.

B

- 2.** The industry standard for LANs.

E

3. A node that handles communication between its LAN and other networks.
C
4. A network that connects other networks.
B
5. Star technology is a _____ configuration.
A
6. Ethernet uses _____.
D

For Exercises 7–15, match the word or acronym with the definition or the appropriate blank.

- A. DLS
 - B. TCP/IP
 - C. UDP
 - D. IP
 - E. TCP
 - F. Broadband
7. _____ and voice communication can use the same phone line.
A
 8. DLS and cable modems are _____ connections.
F
 9. An Internet connection made using a digital signal on regular phone lines.
A
 10. Network technologies that generally provide data transfer speeds greater than 128Kbps.
F
 11. The network protocol that breaks messages into packets, reassembles them at the destination, and takes care of errors.
E
 12. The suite of protocols and programs that support low-level network communication.
B
 13. An alternative to TCP that achieves higher transmission speeds.
C
 14. Software that deals with the routing of packets.
D

15. _____ has more reliability than UDP.

E

For Exercises 15–20, match the protocol or standard with what it specifies or defines.

- A. SMTP
- B. FTP
- C. Telnet
- D. HTTP
- E. MIME type

16. Transfer of electronic mail.

A

17. Log into a remote computer system.

C

18. Transfer files to and from another computer.

B

19. Format of email attachments.

E

20. Exchange of World Wide Web documents.

D

For Exercises 21–25, mark the answers true and false as follows:

- A. True
- B. False

21. A port is a numeric designation that corresponds to a particular high-level protocol.

A

22. A firewall protects a local area network from physical damage.

B (it protects it from inappropriate access)

23. Each company can establish its own access control policy.

A

24. Some top-level domains are based on the country in which the registering organization is based.

A

25. Two organizations cannot have the same name for a computer.

B

Exercises 26–63 are problems or short answer questions.

26. What is a computer network?

A computer network is a collection of computing devices connected so that they can communicate and share resources.

27. How are computers connected together?

The computers in a network can be physically connected by wires or cables or logically connected by radio waves or infrared signals.

28. To what does the word *node* (*host*) refer?

A node or host is any addressable device attached to a network.

29. Name and describe two key issues related to computer networks.

Data transfer rate: The speed with which data is moved across the network

Protocol: The set of rules that define how data is formatted and processed across a network

30. What is a synonym for data transfer rate?

Bandwidth

31. Describe the client/server model and discuss how has it has changed how we think about computing.

The client/server is a model in which resources are spread across the Web. The client makes a request for information or an action from a server and the server responds. For example, a file server, a computer dedicated to storing and managing files for network users, responds to requests for files. A web server, a computer dedicated to responding to requests for web pages, produces the requested page. Before the client/server model was developed, a user thought of computing within the boundaries of the computer in front of him or her. Now the functions that were provided within one computer are distributed across a network, with separate computers in charge of different functions.

32. Just how *local* is a local-area network?

A local-area network connects a relatively small number of machines in a relatively close geographical area, usually within the same room or building, but occasionally a LAN spans a few close buildings.

33. Distinguish between the following LAN topologies: ring, star, and bus.

A ring topology is one in which the nodes are connected in a closed loop. A star topology is one in which the nodes are all connected to a central node. A bus topology is one in which the nodes share a common line.

- 34.** How does the shape of the topology influence message flow through a LAN?

In a ring topology, messages flow in only one direction around the LAN. In a star topology, messages flow through the central node. In a bus topology, messages flow in both directions along the bus.

- 35.** What is a MAN and what makes it different from a LAN and a WAN?

A MAN is a metropolitan-area network. It is a network with some of the features of both a LAN and a WAN. Large metropolitan areas have special needs because of the volume of traffic. MANs are collections of smaller networks but are implemented using such techniques as running optical fiber cable through subway tunnels.

- 36.** Distinguish between the Internet backbone and an Internet service provider (ISP).

The Internet backbone is a set of high-speed networks that carry Internet traffic. An ISP is a company that provides access to the Internet, usually for a fee. An ISP connects directly to the Internet backbone or to a larger ISP with a connection to the backbone.

- 37.** Name at least two national ISPs.

AOL and Prodigy are two national ISPs.

- 38.** Name and describe three technologies for connecting a home computer to the Internet.

Phone modem: A modem is a device that converts computer data into an analog audio signal and back again, thus allowing you to transfer data to and from a computer using your telephone line.

DSL line: A DSL (digital subscriber line) is an Internet connection made using digital signals on regular phone lines.

Cable Modem: A cable modem is a device that allows computer network communication using the cable TV connection.

- 39.** What role do ISPs play with the three technologies in Exercise 38?

Each of the technologies in Exercise 38 requires the connection to go through an ISP. With a phone modem, you dial up a computer that is permanently connected to the Internet. Once the connection is made, you may transfer data. A DSL line maintains an active connection between your home and the ISP. The communication set up to and from your home using cable goes through an ISP.

- 40.** What are the advantages and disadvantages of each of the technologies in Exercise 38?

Phone modems are the cheapest because the phone lines are in place, but transfer speed is very slow because computer data must be converted into an analog audio signal for transfer.

DSL service uses regular phone lines to transfer digital data and you do not have to dial in, but you must be within a certain distance of special equipment or the signal degrades.

Cable modems use a service that many people already have, but the signal deteriorates if too many people in the neighborhood have the service.

Both DSL and cable modems are broadband connections.

- 41.** Phone modems and digital subscriber lines (DSL) use the same kind of phone line to transfer data. Why is DSL so much faster than phone modems?

Phone modems translate digital signals to analog in order to send them over voice frequencies. DSL sends the digital signals over the same phone line but at a different frequency. Because DSL and voice are at different frequencies, they can share the same phone line.

- 42.** Why do DSL and cable modem suppliers use technology that devotes more speed to downloads than to uploads?

Users spend more time asking for data to be sent to their machines (downloads) than they do sending data to other machines (uploads). Therefore, DSL and cable modem suppliers maximize the speed on the most common task.

- 43.** Messages sent across the Internet are divided into packets. What is a packet and why are messages divided into them?

A packet is a unit of data sent across a network. It is more efficient to send uniform sized messages across the Internet.

- 44.** Explain the term *packet switching*.

Packets that make up a message are sent individually over the Internet and may take different routes to their destination. When all the packets arrive at the destination they are reassembled into the original message.

- 45.** What is a *router*?

A router is a network device that directs packets between networks toward their final destinations.

46. What is a *repeater*?

A repeater is a network device that strengthens and propagates a signal along a lone communication line.

47. What problems arise due to packet switching?

Because packets may take different routes, they may not arrive in order. Thus, they must be reassembled into the right order at the receiving end.

48. What are *proprietary systems* and why do they cause a problem?

A proprietary system is one designed and built by a commercial vendor that keeps the technologies used private. If a network's software is a proprietary system, then it can only communicate with other networks that use the same software.

49. What do we call the ability of software and hardware on multiple platforms from multiple commercial vendors to communicate?

Interoperability

50. What is an *open system* and how does it foster interoperability?

An open system is a system based on a common model of network architecture adhering to an accompanying suite of protocols. If all commercial vendors adhere to a common logical architecture and protocols, then networks on multiple platforms from multiple vendors can communicate.

51. Compare and contrast proprietary and open systems.

Both proprietary and open systems can be used to create networks. Networks using the same proprietary systems can communicate with each other, but not with networks that do not use the same system. Networks using open systems can all communicate.

52. What is the seven-layer logical breakdown of network interaction called?

Open Systems Interconnection (OSI) Reference Model

53. What is a *protocol stack* and why is it layered?

A protocol stack is layers of protocols that build and rely on each other. Protocols are layered so that new protocols can be developed without abandoning fundamental aspects of lower levels.

54. What is a *firewall*, what does it accomplish, and how does it accomplish it?

A firewall is a computer system that protects a network from inappropriate access. A firewall filters incoming traffic, checking the validity of incoming messages, and perhaps denying access to messages. For

example, a LAN might deny any remote access by refusing all traffic that comes in on port 23 (the port for telnet).

- 55.** What is a *hostname* and how is it composed?

A hostname is a unique identification for a specific computer on the Internet made up of words separated by dots.

- 56.** What is an *IP address* and how is it composed?

An IP address is made up of four numeric values separated by dots that uniquely identifies a computer on the Internet.

- 57.** What is the relationship between a hostname and an IP address?

Hostnames are for people and IP addresses are for computers. Each hostname is translated into a unique IP address. People refer to the machine by its hostname; computers refer to the machine by its IP address.

- 58.** Into what parts can an IP address be split?

An IP address can be split into a network address, which specifies the network, and a host number, which specifies a particular machine on the network.

- 59.** What are the relative sizes of Class A networks, Class B networks, and Class C networks?

Class A networks are the largest, Class B networks are in the middle, and Class C networks are the smallest.

- 60.** How many hosts are possible in Class C networks, in Class B networks, and in Class A networks?

Class C networks use three bytes for the network number and only one byte for the host number, so they can identify 256 hosts. Class B networks use two bytes for the network number and two bytes for the host number, so they can identify 32768 hosts. Class A networks use one byte for the network number and three bytes for the host number, so they can identify 2^{24} hosts.

- 61.** What is a *domain name*?

A domain name is that part of the hostname that specifies the organization or group to which the host belongs.

- 62.** What is a *top-level domain name*?

The last part of a domain name that specifies the type of organization or its country of origin.

- 63.** How does the current domain name system try to resolve a hostname?

First a request is sent to a nearby domain name server (a computer that attempts to translate a hostname into an IP address). If that

server cannot resolve the hostname, it sends a request to another domain name server. If the second server can't resolve the hostname, the request continues to propagate until the hostname is resolved or the request expires because it took too much time.

Chapter 16 Exercises

For Exercises 1–10, mark the answers true and false as follows:

A. True

B. False

1. The Internet and the Web are essentially two names for the same thing.

B

2. The computer that is set up to respond to web requests is a web browser.

B (it's a web server)

3. When we visit a website, we actually bring the site to us.

A

4. Most search engines use a context-based approach for finding candidate pages.

B

5. A cookie is a program that is executed on your computer.

B (most use key word matching)

6. All elements associated with a particular web page are brought over when a request for that web page is made.

A

7. Networks have been used to connect computers since the 1950s.

A

8. Network communication was not possible until the advent of the Web.

B

9. The Web was developed in the mid-1990s.

A

10. You must have a web browser in order to access the Web.

A

For Exercises 11–20, match the word or acronym with the definition or blank.

- A. JSP scriptlet
- B. URL
- C. HTML
- D. Tag
- E. Java applet
- F. XML

11. A program designed to be embedded into an HTML document.
E
12. Uniquely identifies every web page.
B
13. _____ runs on the web server.
A
14. _____ runs on the web browser.
E
15. Tags in _____ are fixed.
C
16. Tags in _____ are not predefined.
F
17. _____ is a metalanguage.
F
18. The structure of an _____ document is described by its corresponding Document Type Definition (DTD).
F
19. The syntactic element in a markup language that indicates how information should be displayed.
D
20. Part of a _____ is the host name of the computer on which the information is stored.
B

Exercises 21–68 are problems or short answer questions.

21. What is the Internet?
The Internet is a wide-area network spanning the globe.
22. What is the Web?
The Web is an infrastructure of distributed information combined with the software that uses networks as a vehicle to exchange that information.

23. What is a web page?

A web page is a document that contains or references various kinds of data such as text, images, graphics, and programs.

24. What is a website?

A website is a collection of related web pages usually designed and controlled by the same person or company.

25. What is a link in the context of the Web?

A link is a connection between one web page and another.

26. Why is a spider web a good analogy for the World Wide Web?

The Internet is the hardware upon which the spider-like connections of the World Wide Web have been created.

27. What is the relationship between a web page and a website?

A web page is a document that contains or references various kinds of data. A website is a collection of related web pages.

28. What is the difference between the Internet and the Web?

The Internet is a wide area network that spans the earth. The Web is the infrastructure of distributed information and network software that lets us use the Internet more easily.

29. Describe how a web page is retrieved and viewed by a web user.

When a web address is specified in a browser, the browser sends a request to that site. The site receiving the request sends the page and all associated information back to be displayed in the browser.

30. What is a Uniform Resource Locator?

A Uniform Resource Locator (URL) is the standard way of specifying the location of a web page.

31. What is a markup language? Where does the name come from?

A markup language is one that uses tags to identify the elements in a document and indicate how they should be displayed. The name comes from the idea of taking a document and writing (marking up) the document with tags that say how to display it.

32. Compare and contrast hypertext and hypermedia.

Hypertext and hypermedia both mean that data (information) is not organized linearly. There are embedded links that allow us to jump from one place to another in documents. Because information on the Web is more than just text, hypermedia is a more accurate term.

33. Describe the syntax of an HTML tag.

HTML tags are composed of reserved words enclosed in angled brackets (<...>). Some reserved words are used in pairs with the second one preceded by a /.

- 34.** What is a horizontal rule? What are they useful for?

Horizontal rules are lines across a page. They are useful for separating sections of a page.

- 35.** Name five formatting specifications that can be established using HTML tags.

HTML is not case sensitive.

`.. bold`

`<i>..</i> italic`

`<hr>` horizontal rule

`.. unordered list`

`.. ordered list`

`` list item

`<h3>..</h3> number 3 heading`

- 36.** What is a tag attribute? Give an example.

A tag attribute is part of a tag that gives extra information. `` is an example. The tag is *img* for image, and the attribute is *src* for source of image, which is followed by the name of a file containing the image in quotes.

- 37.** Write the HTML statement that inputs the image on file "mine.gif" into the web page.

``

- 38.** Write the HTML statement that sets up a link to `http://www.cs.utexas.edu/users/ndale/` and shows the text "Dale Home Page" on the screen.

` Dale Home Page `

- 39.** What happens when a user clicks on "Dale Home Page" as set up in Exercise 38?

A copy of the page at `http://www.cs.utexas.edu/users/ndale` is displayed on the user's browser.

- 40.** Design and implement an HTML document for an organization at your school.

Activity, no answer expected.

- 41.** Design and implement an HTML document describing one or more of your personal hobbies.

Activity, no answer expected.

42. What is a Java applet?

A Java applet is a Java program designed to be embedded in an HTML document, transferred over the Web, and executed in a browser.

43. How do you embed a Java applet in an HTML document?

A Java applet is embedded in an HTML document using the applet tag. For example, the following HTML tag embeds class `MyClass` in an HTML document:

```
<APPLET code="MyClass.class" width=250 height=150 ></APPLET>
```

44. Where does a Java applet get executed?

A Java applet gets executed in the user's browser.

45. What kinds of restrictions are put on Java applets? Why?

Because a Java applet is executed on the user's machine, it must be transmitted from the web server. Also, the user's computer may not have a resource that the applet needs. Thus only relatively small programs using very standard resources are appropriate.

46. What is a Java Server Page?

A Java Server Page (JSP) is a web page that has Java scriptlets embedded in it.

47. What is a scriptlet?

A scriptlet is a code segment embedded in an HTML document designed to contribute to the content of the page.

48. How do you embed a scriptlet in an HTML document?

The special HTML tags `<%..%>` enclose the scriptlet.

49. How does JSP processing differ from applet processing?

Scriptlet processing is done on the server side; applet processing is done on the user's side.

50. What is a metalanguage?

A metalanguage is a language used to define other languages.

51. What is XML?

XML is a metalanguage that is used to define other markup languages.

52. How are HTML and XML alike and how are they different?

Both HTML and XML used tagged data. The tags that HTML uses are predefined, both in terms of syntax and semantics. XML is a metalanguage that defines new markup languages. An XML document is written using tags, which are then defined in the accompanying Document Type Definition.

- 53.** How does an XML document relate to a Document Type Definition?
The XML document and the related DTD define a new markup language.
- 54. a.** In a DTD, how do you indicate that an element is to be repeated zero or more times?
An element in parentheses with an asterisk following the element indicates zero or more times.
- b.** In a DTD, how do you indicate that an element is to be repeated one or more times?
An element in parentheses with a plus sign following the element indicates one or more times.
- c.** In a DTD, how do you indicate that an element cannot be broken down into other tags?
An element followed by (#PCDATA) indicates that the element can not be broken down further.
- 55.** What is XSL?
XSL stands for Extensible Style sheet Language. XSL is used to define transformations of XML documents to other formats.
- 56.** What is the relationship between XML and XSL?
XSL is the language that can be used to determine formats for the XML document and its accompanying DTD.
- 57.** How does an XML document get viewed?
An XML document is translated by XSL into a form that can be displayed.
- 58.** Define an XML language (the DTD) for your school courses and produce a sample XML document.
- ```
<?xml version="1.0" ?>
<!DOCTYPE courses SYSTEM "courses.dtd">
<courses>
 <course>
 <title>Analysis of Algorithms</title>
 <department>Computer Science</department>
 <courseNumber>170</courseNumber>
 <instructor>Smyth</instructor>
 </course>
</courses>
<title>The American Revolution</title>
```

```

<department>History</department>
<courseNumber>240</courseNumber>
<instructor>Jefferson</instructor>
</course>
<courses>
<title>Spanish Poetry</title>
<department>Spanish</department>
<courseNumber>470</courseNumber>
<instructor>Garcia</instructor>
</course>
</courses>

```

```

<!ELEMENT courses (course*) >
<!ELEMENT course (title, department, courseNumber, instructor)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT department (#PCDATA)>
<!ELEMENT courseNumber (#PCDATA)>
<!ELEMENT instructor (#PCDATA)>

```

- 59.** Define an XML language (the DTD) for political offices and produce a sample XML document.

```

<?xml version="1.0" ?>
<!DOCTYPE government SYSTEM "government.dtd">
<government>
<position>
<title>President of the United States</title>
<type>Federal</type>
<currentHolder>
<name>George W. Bush</name>
<party>Republican</party>
</currentHolder>
<pastHolders>
<name>William Clinton</name>
<name>George H. W. Bush</name>
<name>Ronald Reagan</name>

```



```

 <name>James Carter</name>
 </pastHolders>
</position>
<position>
 <title>Vice President of the United States</title>
 <type>Federal</type>
 <currentHolder>
 <name>Richard Cheney</name>
 <party>Republican</party>
 </currentHolder>
 <pastHolders>
 <name>Al Gore</name>
 <name>Dan Quayle</name>
 <name>George H. W. Bush</name>
 <name>Walter Mondale</name>
 </pastHolders>
</position>
</government>

<!--ELEMENT government (position*) -->
<!--ELEMENT position (title, type, currentHolder, pastHolders)-->
<!--ELEMENT title (#PCDATA)-->
<!--ELEMENT type (#PCDATA)-->
<!--ELEMENT currentHolder (name, party)-->
<!--ELEMENT pastHolders (name*)-->
<!--ELEMENT name (#PCDATA)-->
<!--ELEMENT party (#PCDATA)-->

```

**60.** Define an XML language (the DTD) for zoo animals and produce a sample XML document.

```

<?xml version="1.0" ?>
<!DOCTYPE animals SYSTEM "animals.dtd">
<animals>
 <animal>
 <commonName>kangaroo</commonName>

```

```
<class>mammalia</class>
<order>marsupialia</order>
<onSite>
 <male>
 <number>2</number>
 <name>Cletus</name>
 <name>Nate</name>
 </male>
 <female>
 <number>0</number>
 </female>
</onSite>
</animal>
<animal>
 <commonName>elephant</commonName>
 <class>mammalia</class>
 <order>elephantidae</order>
 <onSite>
 <male>
 <number>1</number>
 <name>Max</name>
 </male>
 <female>
 <number>2</number>
 <name>Beauty</name>
 <name>Geraldine</name>
 </female>
 </onSite>
</animal>
<animal>
 <commonName>alligator</commonName>
 <class>reptilia</class>
 <order>crocodilia</order>
 <onSite>
```

```

<male>
 <number>4</number>
</male>
<female>
 <number>7</number>
</female>
</onSite>
</animal>
</animals>

```

```

<!ELEMENT animals (animal*) >
<!ELEMENT animal (commonName, class, order, onSite)>
<!ELEMENT commonName (#PCDATA)>
<!ELEMENT class (#PCDATA)>
<!ELEMENT order (#PCDATA)>
<!ELEMENT onSite (male, female)>
<!ELEMENT male (number, name*)>
<!ELEMENT female (number, name*)>
<!ELEMENT number (#PCDATA)>
<!ELEMENT name (#PCDATA)>

```

**61.** This chapter is full of acronyms. Define each of the following ones.

- a.** HTML  
Hypertext Markup Language
- b.** XML  
Extensible Markup Language
- c.** DTD  
Document Type Definitions
- d.** XSL  
Extensible Stylesheet Language
- e.** SGML  
Standard Generalized Markup Language
- f.** URL  
Uniform Resource Locator
- g.** ISP  
Internet Service Providers

- 62.** Create an HTML document for a web page that has each of the following features.
- a.** centered title
  - b.** unordered list
  - c.** ordered list
  - d.** link to another web page
  - e.** a picture
- Activity; no answer expected.
- 63.** Distinguish between an HTML tag and an attribute.
- A tag is a syntactic element in a markup language that indicates how information should be displayed. An attribute is part of a tag that gives additional information about it.
- 64.** Why does the same web page look different in different browsers?
- The tags in the HTML document that defines a web page may be interpreted differently by different browsers.
- 65.** What are the two sections of every HTML document?
- The head of the document and the body of the document.
- 66.** What are the contents of the two parts of an HTML document?
- The head contains information about the document. The body contains the information to be displayed.
- 67.** What does the A stand for in the tag that specifies a URL for a page?
- Anchor
- 68.** Create an HTML document for a web page that has each of the following features.
- a.** a right-justified title in large type font
  - b.** an applet class named “Exercise.class”.
  - c.** two different links
  - d.** two different pictures
- Activity; no answer provided.

## Chapter 17 Exercises

For Exercises 1–15, match the Big-O notation with its definition or use.

- A.**  $O(1)$
- B.**  $O(\log_2 N)$
- C.**  $O(N)$
- D.**  $O(N \log_2 N)$
- E.**  $O(N^2)$
- F.**  $O(2^N)$
- G.**  $O(N!)$

1. Factorial time  
G
2.  $N \log N$  time  
D
3. Linear time  
C
4. Quadratic time  
E
5. Exponential time  
F
6. Logarithmic time  
B
7. Bounded time  
A
8. Time not dependent on the size of the problem.  
A
9. Algorithms that successively cut the amount of data to be processed in half at each step.  
B
10. Mergesort and Heapsort  
D
11. Selection sort and Bubble sort  
E
12. Adding a column of  $N$  numbers  
C
13. Demonstrated by the fable of the King and the Corn.  
F
14. Traveling salesperson  
G
15. What Quicksort degenerates to if the data is already sorted.  
E

For Exercises 16–20, match the name of the technique with the algorithm.

- A.** Even parity
- B.** Odd parity
- C.** Check digits
- D.** Error-correcting codes
- E.** Parity bit

**16.** An extra bit associated with each byte in the hardware that ensures that the number of 1 bits is odd or even across all bytes.

E

**17.** Ultimate redundancy would be to keep two copies of every value.

D

**18.** The number of 1 bits plus the parity bit is odd.

B

**19.** The number of 1 bits plus the parity bit is even.

A

**20.** A scheme to sum the individual digits in a number and store the unit's digit of that sum with the number.

C

For Exercises 21–30, mark the answers true and false as follows:

- A.** True
- B.** False

**21.**  $(1 + X - 1)$  is always equal to  $X$ .

B

**22.** Representational error is a synonym for round-off error.

A

**23.** Software verification activities are limited to the implementation phase.

B

**24.** Half the errors in a software project occur in the design phase.

A

**25.** Most large software projects are designed by a single genius and then given to teams of programmers to implement.

B

**26.** The later in the software life cycle that an error is detected, the cheaper it is to fix.

B

- 27.** Formal verification of programs is of theoretical interest but has so far never been useful.  
B
- 28.** Big-O notation tells us how long the solution takes to run in terms of microseconds.  
B
- 29.** Software engineering, a branch of computing, emerged in the 1960s.  
A
- 30.** Maintaining and evolving existing software has become more important than building new systems.  
A

Exercises 31–61 are problems or short answer questions.

- 31.** Define representational error, cancellation error, underflow, and overflow. Discuss how these terms are interrelated.

Representational error (round-off error) is the error caused by the fact that the precision of the result of an arithmetic operation is greater than the precision of our machine.

Cancellation error is the loss of accuracy during addition or subtraction of numbers of widely differing sizes, due to limits of precision.

Underflow is the condition that occurs when the results of a calculation are too small to represent in a given machine.

Overflow is the condition that occurs when the results of a calculation are too large to represent in a given machine.

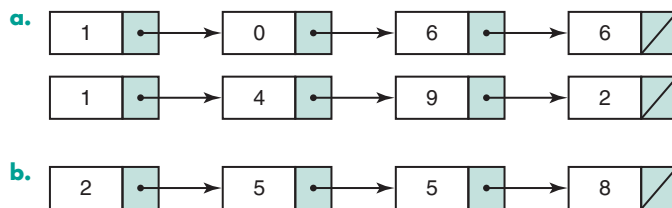
These terms are related because they all refer to problems that occur because numbers are infinite, and their representation in a computer is finite.

- 32.** Show the range of integer numbers that can be represented for each of following word sizes.
- a.** 8-bits  
-128..127
  - b.** 16 bits  
-32768..31767
  - c.** 24 bits  
-8388608..8388607
  - d.** 32 bits  
-2147483648..2147483647
  - e.** 64 bits  
-9223372036854775808..9223372036854775807

33. There is a logical action to take when underflow occurs, but not when overflow occurs. Explain.

Underflow occurs when the number is too small to represent. When this happens, it is logical to set the value to 0. Overflow occurs when the number is too big to represent. There is no logical value to substitute when this occurs.

34. a. Show how the numbers 1066 and 1492 would be represented in a linked list with one digit per node.  
 b. Use a linked list to represent the sum of these integers.  
 c. Outline an algorithm to show how the calculation might be carried out in a computer.



- c. To calculate the sum, you must move from right to left in the list rather than left to right. Assume a *previous* operation gets the node before *current*. *MOD* is an operation that returns the remainder from integer division. *DIV* is an operation that returns the quotient from integer division.

```

Set currentFirst to last
Set currentSecond to last
Set carry to 0
While (currentFirst <> NULL and currentSecond <> NULL)
 Get a new node
 Set info(new node) to (info(currentFirst) + info(currentSecond) +
 carry) MOD 10
 Set carry to (info(currentFirst) + info(currentSecond) + carry)
 DIV 10
 Set currentFirst to previous(currentFirst)
 Set currentSecond to previous(currentSecond)
 Put new node into result

```



```

While (currentFirst <> NULL)
// Copy rest of first list if it is not empty.
 Get a new node
 Set info(new node) to (info(currentFirst) + carry) MOD 10
 Set carry to (info(currentFirst) + carry) DIV 10
 Set currentFirst to previous(currentFirst)
 Put new node into result

While (currentSecond <> NULL)
// Copy rest of second list if it is not empty
 Get a new node
 Set info(new node) to (info(currentSecond) + carry) MOD 10
 Set carry to (info(currentSecond) + carry) DIV 10
 Set currentSecond to previous(currentSecond)
 Put new node into result

```

- 35.** Explain the Titanic Effect in relation to hardware failure.

The Titanic Effect states that the severity with which a system fails is directly proportional to the intensity of the designer's belief that it cannot. The more intense the designer's belief is that the product cannot fail, the more catastrophic a failure is.

- 36.** Have any hardware failures happened to you? Explain.

No answer expected.

- 37.** Given the following 8-bit code, what is the parity bit if odd parity is being used?

**a.** 11100010

1

**b.** 10101010

1

**c.** 11111111

1

**d.** 00000000

1

**e.** 11101111

0

- 38.** Given the following 8-bit code, what is the parity bit if even parity is being used.
- a.** 11100010  
0
  - b.** 10101010  
0
  - c.** 11111111  
0
  - d.** 00000000  
0
  - e.** 11101111  
1
- 39.** Given the following numbers, what would be the check digit for each?
- a.** 1066  
3
  - b.** 1498  
2
  - c.** 1668  
1
  - d.** 2001  
3
  - e.** 4040  
8
- 40.** What errors would be detected using the check digits in Exercise 39?  
*This technique recognizes when one digit of a number is corrupted.*
- 41.** Given the following numbers, what would be the additional digits if the unit's digit of the sum of the even digits is used along with the check digit?  
*Counting is from left to right.*
- a.** 1066  
6
  - b.** 1498  
2
  - c.** 1668  
4
  - d.** 2001  
1
  - e.** 4040  
0

- 42.** Given the following numbers, what would be the additional digits if the unit's digit of the sum of the odd digits is used along with the check digit?

Counting is from left to right.

**a.** 1066

7

**b.** 1498

0

**c.** 1668

7

**d.** 2001

2

**e.** 4040

8

- 43.** How do the representations in Exercises 41 and 42 improve the error detection over a simple check digit?

This technique catches a transposition error between adjacent digits.

- 44.** Explain the concept of the software life cycle.

The software life cycle is the concept that software is developed, not just coded, and evolves over its lifetime.

- 45.** Where do most of the errors occur in a software project?

Half the errors occur in the design phase and half occur in the implementation phase.

- 46.** Why does the cost of fixing an error increase the longer the error remains undetected?

Software evolves over time. When a error is detected it must be corrected where it first occurs and every subsequent place to which the error propagates.

- 47.** Compare and contrast the software verification activities code to design walk-throughs and inspections.

A walk-through is an activity in which a team performs a manual simulation of the program or design. An inspection is an activity in which one member of a team reads the program or design line by line and the others point out errors. Both are group activities, but an inspection is lead by one person.

- 48.** How can a program be verified to be correct but still be worthless?

A program may do exactly as its specification requires, but if the specification is incorrect, the program is worthless.

- 49.** Name at least 5 places where a software error could be introduced.

Errors can be introduced in each phase of the software life cycle: requirements, specifications, design, implementation, and maintenance.

- 50.** How was the AT&T software failure typical of such failures?

One person made the error, but many others reviewed the code without seeing the error. The failure was triggered by a relatively uncommon sequence of events that were difficult to anticipate in advance. The error occurred in code designed to improve a correctly working system.

- 51.** What is formal verification?

Formal verification is the verification of program correctness independent of testing. The goal is to develop a method for proving programs that is analogous to the method of proving theorems in geometry.

- 52.** Explain the analogy of the elephant and the goldfish.

The relative size of the elephant dwarfs the contribution of the goldfish.

- 53.** Define polynomial time.

A polynomial time algorithm is one whose complexity can be expressed as a polynomial in the size of the problem.

- 54.** How is it possible to throw away all but the term with the largest exponent when assessing the Big-O of a polynomial time algorithm?

As the size factor increases, the largest term dwarfs the contribution of the other terms.

- 55.** Give the Big-O complexity measure of the following polynomials.

**a.**  $4x^3 + 32x^2 + 2x + 1003$

$O(N^3)$

**b.**  $x^5 + x$

$O(N^5)$

**c.**  $x^2 + 124578$

$O(N^2)$

**d.**  $x + 1$

$O(N)$

- 56.** Explain the analogy of bins of complexity measures.

We can think of a bin representing one of the Big-O complexity measures. The bin contains all of the algorithms that have that complexity measure, but within the bin the algorithms can be ordered by the discarded terms.

- 57.** Who manufactures a Turing machine?

A Turing machine is a hypothetical machine used as a model to study the limits of what can be computed.

- 58.** How does a Turing machine simulate a human with a paper and pencil?

A Turing Machine consists of a control unit with a read/write head (the brain) that can read and write symbols on an infinite tape. The tape is divided into cells. The model is based on a person doing a primitive calculation on a long strip of paper using a pencil with an eraser. Each line (cell) of the paper contains a symbol from a finite alphabet. Starting at one cell, the symbol is examined and either left alone or erased and replaced with another symbol from the alphabet. An adjacent cell is accessed and the actions are repeated.

- 59.** Are there problems for which there are no solutions?

Yes; there is at least one.

- 60.** Describe the Halting Problem.

The Halting Problem asks the question: Given a program and an input to the program, determine if the program will eventually stop with this input.

- 61.** How is the fact that data and programs look alike inside a computer used in the proof that the Halting problem is unsolvable?

In the proof, the program is given itself as data.

