Study questions on Topic 1: JavaScript

1. What does the following JavaScript code do?
   (a) var a, b, total
   (b) x = parseInt(prompt(“Enter your age”, “”))
   (c) alert(“Your age is “ + x)
2. Describe two meanings of “+” in JavaScript.
3. Contrast machine language to JavaScript.
4. Contrast JavaScript to HTML.
5. Describe how JavaScript and HTML may be used in the same web-resident file.
6. Circle the JavaScript declaration statement below. [show code]
7. Write a JavaScript statement that displays a rectangle with the word “Goodbye” in it and an “OK” button.
8. What is an event handler?
9. Compare the languages of expressions used in Excel formulas and JavaScript code.

Multiple-choice:
1. JavaScript encodes (a) web-page formatting; (b) responses to input events; (c) database design; (d) statistical analysis; (e) none of these
2. JavaScript code is likely to appear in (a) spreadsheet formulas; (b) processor registers; (c) database queries; (d) HTML files; (e) none of these
3. Variables may be assigned values in (a) JavaScript statements; (b) spreadsheet formulas; (c) named styles; (d) packet transmission; (e) none of these
4. Of the following, which is a high level programming language? (a) assembler; (b) HTML; (c) machine; (d) JavaScript; (e) none of these
5. In JavaScript, “+” signifies (a) addition; (b) concatenation; (c) addition and concatenation; (d) any operation; (e) none of these
6. An event handler is (a) a program; (b) program code; (c) a browser; (d) a job category; (e) none of these
7. `parseInt` (a) is a variable; (b) converts strings to integers; (c) converts numbers to strings; (d) is a language; (e) none of these
8. (a) ; (b) ; (c) ; (d) ; (e)
9. (a) ; (b) ; (c) ; (d) ; (e)
10. (a) ; (b) ; (c) ; (d) ; (e)
11. 
Study questions on Topic 2 (Hardware)

Multiple-choice or T/F:
1. Which instruction below does not change the value in the accumulator? (a) load; (b) store; (c) add; (d) sub
2. Which is not a peripheral? (a) the microprocessor; (b) the monitor; (c) the keyboard; (d) the printer; (e) a disk drive
3. The binary system is appropriate for digital computers because (a) our hands have ten fingers; (b) binary arithmetic is simpler by nature; (c) binary may be easily translated to hexadecimal notation; (d) binary and decimal values are easily converted; (e) digital computers are based on two-state devices
4. Input/output is data that moves between RAM and (a) the program counter; (b) data registers; (c) the instruction register; (d) RAM; (e) peripherals
5. 0011₂ + 0010₂ = (a) 0101₂; (b) 0100₂; (c) 1000₂; (d) 0111₂; (e) 1011₂
6. (T-F) A yes-or-no answer to a question must be stored in at least three bytes of memory.
7. A data statement in an assembler program introduces (a) a memory address by number; (b) a memory address using a name; (c) the accumulator; (d) the program counter; (e) a user-input value
8. Electronic storage composed of silicon chips is (a) RAM; (b) CDROM; (c) hard disk; (d) keyboard; (e) monitor
9. Microprocessors have on them: (a) a disk; (b) a screen; (c) registers; (d) high-level code; (e) documentation
10. (T-F) A microprocessor has its own special assembler language.
11. A language expressed in binary notation is (a) machine language; (b) assembler language; (c) Java; (d) all of the above; (e) none of the above
12. The characteristic feature of all general-purpose computers is that they (a) display colors; (b) have CDROMs; (c) can be upgraded; (d) can store programs; (e) run faster than 166 MHz.
13. A NOT gate is a(n) (a) software building block; (b) hardware building block; (c) design tool; (d) algorithm; (e) Java operator
14. (T-F) A black-box diagram shows both what a component does and how.
15. 3 = (a) 111₂; (b) 11₂; (c) 10₂; (d) 11000₂; (e) none of these
16. (T-F) The fetch-execute cycle runs until the user inputs a Quit instruction.
17. A disk drive is a (a) software component; (b) register; (c) peripheral; (d) logic gate; (e) silicon-based storage device
18. (T-F) The operating system runs concurrently as applications execute.
19. A server is (a) a computer; (b) a peripheral; (c) an application program; (d) a program that runs on all network workstations; (e) an agent
20. The hexadecimal system uses base (a) 2; (b) 4; (c) 8; (d) 16; (e) 60
21. The OR gate (a) is a peripheral; (b) contains a register; (c) yields a 0 if both its inputs are 1; (d) yields a 0 unless both its inputs are 1; (e) produces a 1 if either of its inputs is 1
22. The AND gate (a) is a peripheral; (b) contains a register; (c) yields a 0 if both its inputs are 1; (d) yields a 0 unless both its inputs are 1; (e) produces a 1 if either of its inputs is 1
23. A one-input circuit that outputs a 1 on input of 0 and a 0 on input of 1 is (a) NOT; (b) OR; (c) AND; (d) MAYBE; (e) XOR
24. (T-F) A bitwise AND operation corresponds to a simple arithmetic procedure taught in grade school.
25. (T-F) The bitwise complement operation (~) corresponds to subtracting a number from zero.
26. 11₂ + 101₁₂ = (a) 0011₂; (b) 0100₂; (c) 1000₂; (d) 0111₂; (e) 1110₂
27. 1110₂ + 101₂ = (a) 0011₂; (b) 1010₂; (c) 1001₂; (d) 0111₂; (e) 1011₂
28. Machine language is (a) easier to read than C or Java; (b) the language of the executable program; (c) the language of the source code; (d) the compiler’s input; (e) none of these
29. The architecture of a computer is its (a) brand name; (b) application programs; (c) operating system; (d) hardware organization; (e) software design
30. Place values are used in (a) memory; (b) software design; (c) decimal numbers only; (d) binary, decimal, and hexadecimal numerals; (e) assembler programs

31. In the binary-to-decimal conversion presented, there is one step for each (a) value converted; (b) decimal digit; (c) binary digit; (d) remainder; (e) carry

32. In the decimal-to-binary conversion presented, there is one step for each (a) value converted; (b) decimal digit; (c) binary digit; (d) remainder; (e) carry

33. \(1001_2 + 0011_2 = (a) 1011_2; (b) 0100_2; (c) 0101_2; (d) 1000_2; (e) 0011_2\)

34. Negative binary values are stored using (a) overflow; (b) accumulator; (c) carry; (d) borrow; (e) a sign bit

35. When a computation produces a value greater than the hardware’s capacity to store it, what is the result? (a) caching; (b) overflow and loss of data; (c) program termination; (d) complementing; (e) compression

36. Floating-point representation adds ____ to what is found in integer storage. (a) a sign bit; (b) an exponent; (c) a binary point; (d) flotation; (e) speed

37. “Assembler” is (a) a language adapted both to the hardware and to human use; (b) an algorithm to convert numbers; (c) a high-level language; (d) a component of a microprocessor; (e) part of a network operating system

38. A mnemonic (a) specifies a data location; (b) specifies a data value; (c) specifies a constant; (d) specifies an action; (e) is a language

39. (T-F) The language of the microprocessor is assembler.

40. (T-F) The language of a microprocessor is expressed in 0’s and 1’s.

41. (a) ; (b) ; (c) ; (d) ; (e)

42. (a) ; (b) ; (c) ; (d) ; (e)

43. (a) ; (b) ; (c) ; (d) ; (e)

**Short-answer:**

1. What is a low-level language, associated with a particular microprocessor, that uses binary code to represent instructions and data?
2. Convert 10 (decimal) to binary notation.
3. Convert the value 10012 to decimal notation.
4. What information must a microprocessor have about a data item in RAM in order to assign it a value?
5. A circuit’s output, given a certain combination of inputs, is diagrammed by a(n) _______
6. Shifting left a binary number by one bit produces what result?
7. What is a data location that stores intermediate values while calculating?
8. What is a data location that stores the current instruction being executed?
9. What is a data location that stores the address of the next instruction that will execute?
10. Convert to decimal: 10112
11. Convert to decimal: 01012
12. Convert to decimal: 01102
13. Convert to decimal: 10012
14. Convert to decimal: 11002
15. Convert to binary: 1510
16. Convert to binary: 610
17. 10012 + 00112 = _______
18. 01112 + 01002 = _______
19. 10002 – 01102 = _______
20. 01102 – 00102 = _______
21. What storage technique is used to store negative values?
22. What is the result when a computation produces a value that exceeds the hardware’s storage capacity?
23. What is base-16 notation called?
24. What is a low-level language that uses labels and mnemonics to express programs?
25. Name the loop that a microprocessor is running at all times.
26. Name the part of an assembler-language statement that specifies what action is to be taken.
27. Name the part of an assembler-language statement that specifies the location of the data to be acted upon.
28. Name the part of an assembler-language statement that specifies the name of the statement’s address.
29. How many different values can be represented in 16 bits?
30. What information must a microprocessor have about a data item in RAM in order to access it?
31. Name four peripheral devices.
32. What component of a microprocessor stores:
(a) internal results of a computation?
(b) the address of the next instruction to be executed?
(c) the current instruction being executed?
33. What is the common name for primary storage?
34. If one of the sixteen bits of a memory cell were not used as a sign bit, what further restriction would be placed on the type of numbers a 16-bit computer could process?
35. How much data is required to store the information as to whether or not a number is greater than zero?
36. Do each of the following number base conversions:
(a) 23 = ___________2
(b) 12 = ___________2
(c) 10110012 = _______10
(d) 1012 = _______10
(e) 110112 = _______10
(f) 10101112 = _______10
(g) 35 = _______2
(h) 77 = _______2
37. Using the binary storage format outlined in Section 1.5, show how each of the following decimal values would be stored in memory.
(a) 375
(b) –1270  (c) –1 (d) 32,127
38. What decimal values are represented by the following stored binary integer values?
(a) 1000000000011101
(b) 0100000011111000
(c) 1111111111111111
(d) 0111111100000000
39. What component of a microprocessor stores:
(a) internal results of a computation?
(b) address of the next instruction to be executed?
(c) the current instruction being executed?
40. What information must a machine-language program have about a data item in a computer in order to access it?
41. How much data is required to store the answer to a yes/no question? __________
42. What range of values can be stored in 16 bits? __________
43. Do these conversions:
(a) 6 = _______2
(b) 1002 = _______10
(c) 13 = _______2 (d) 101012 = _______10
44. Find the binary results:
(a) 102 + 1012 = __________
(b) 101112 + 11012 = __________
(c) 101002 – 11102 = __________
45. Express each of these in exponential notation:
(a) 0.47 ____________
(b) 392.1 ____________
46. Write the truth table of the NAND gate:
47. Show result of these bitwise operations:
(a) 01012 | 11102 = _______
(b) 10102 >> 1 = ______
48. Label each term below with the letter of its appropriate definition.
____ accumulator
____ assembly language
____ bit
____ instruction register
____ label
____ machine language
____ microprocessor
____ operating system
____ mnemonic
____ operand
____ program counter
____ RAM
____ software
a) Computer programs.
b) Component of a software system
c) Hardware that contains logic for controlling program execution and manipulating data.
d) A binary digit, having a value of either 0 or 1.
e) Electronic data storage that uses silicon chips.
f) Register that holds the program instruction currently being executed.
g) The register in the processor that at any instant specifies the location in memory of the next instruction to be executed.
h) In the processor, the register that stores either data copied from a memory cell or the result of an operation on that data.
i) Software that manages a computer’s activities while other programs run, providing services to programs and to the user.
j) Has one mnemonic for each machine instruction of a processor.
k) Set of binary-coded instructions for a particular microprocessor
l) A data item to be operated on by an instruction
m) A machine instruction expressed in easily remembered word form.

n) A name chosen by the programmer for an address in memory.

49. 

```
load fee
sub discount
store bill
stop
fee data 100
discount data 20
bill data 0
```

In the program above, what is the effect of the line

a) load fee
b) sub discount
c) store bill
d) fee data 100

**Longer answer:**

1. Write a program for the model processor’s assembly language that accepts two values and displays the smaller one.
2. Describe the fetch/execute cycle in one or two paragraphs.
3. In what ways do the human brain and a computer resemble each other? Differ?
4. Why is the binary number system appropriate for digital computers?
5. What is the advantage of using a special two’s-complement format for negative numbers instead of normal binary format?

**Answers to hardware study questions**

13. b. A NOT gate is a logic component built from transistors.
14. f. A black box hides implementation details.
15. b. \(3 = 1 \times 2 + 1 \times 1 = 11_2\)
16. f. The fetch-execute cycle runs until a `stop` instruction is encountered.
17. c. Input/output devices are peripherals.
18. t. The operating system manages resources for the user and for application software.
19. a. A server is a PC that handles data communication in a network and makes applications and data available to network workstations.
20. d. Base-16 notation records four bits per hexadecimal digit.
21. e. The OR gate yields a 0 only if both its inputs are 0.
22. d. The AND gate outputs 1 if both inputs are 1.
23. a. The NOT gate outputs the logical negation of its single input.
24. f. Bitwise AND performs logical conjunction on each bit.
25. f. The bitwise complement of a number is the same number with each bit logically negated, from 1 to 0 or 0 to 1.
26. e. \(11 + 1011 = 3 + 11 = 14 = 1110\)
27. –
28. –
29. –
30. –
31. –
32. –
33. –
Short-answer:
1. Machine language
2. $101_2$
3. $1001_2 = 9$
4. Its address
5. truth table
6. Doubles the number
7. register
8. instruction register
9. program counter
10. $12_{10}$
11. $1011_2 = 11_{10}$
12. $0101_2 = 5_{10}$

Short-answer:
1. What are some significant ways that processor-based systems as used in IT differ from living organisms?
2. Relate processor, RAM, and I/O.
3. Contrast two or three categories of physical media or devices used in data storage.
4. In what way is the binary system of numerals relevant to information technology?
5. Support or refute: All information processed or communicated by information technology takes a common form.
6. In what way may it be said that bits have no meaning, in themselves?
7. How is it possible for a single processor to execute multiple computer programs concurrently?
8. In what sense does an operating system manage memory?
9. Explain how when you use a program like MS Word, you sometimes are making use of the Windows operating system.
10. What is a file?
11. What is a folder?
12. How many bits are there in a kilobyte?
13. Is a shortcut to a folder itself a folder? Explain why or why not.
14. When we speak of your “Y: drive,” in My Computer, are we speaking of a physical disk drive? Explain.
15. What is a dialog box? A pop-up menu?
16. Distinguish radio buttons from check boxes.
17. In what part of an IT system is the arithmetic logic unit found and what else is found there?
18. Relate the following concepts: bits, characters, 32-bit words, bytes, registers, and megabytes.
19. What is a way that analog information is converted to digital form? Digital to analog?
20. Contrast digital and analog devices, giving one example of each.
21. What are some features of a machine language as opposed to a different kind of language?
22. What are the smallest elements of a color image displayed on a monitor, and how are the colors represented?
23. Where is the program counter and what does it do?
24. Give an example of random access and an example of sequential access.
25. Are all the files in a folder located in the same part of a disk? Explain.
26. Convert to decimal: $110$, $1010101$; $1111000$; $110011$; $111010$
27. Convert to binary: $4$
28. Does a bit have an inherent meaning? Why/why not?
29. Distinguish an analog device from a digital one.
30. Order the following according to distance from the user’s perception: operating system, application, hardware
31. Order the following according to distance from the processor control unit: cache, hard drive, RAM, program counter, accumulator
32. What two kinds of components does a port connect?
33. A device driver provides the interface between what kind of software and what kind of hardware?
34. In Windows and some other user interfaces, what does right-clicking on an object enable the user to do?
35. Distinguish the functions of the Backspace and Delete keys.
36. Relate the following: mouse pointer, cursor, Insertion Point.
40. What does the Clipboard store and in what part of the hardware is the Clipboard located?
41. What steps are recommended, in order, if an application stops responding?
42. How do you make a window disappear while leaving it available in the Task Bar?
43. Since Word documents and web sites contain data, why could it be more risky to open a Word document or to click a Web link than to open a text file?
44. What happens if you double-click a file whose name ends, “.exe”?

**Carry out these operations:**
45. Check free disk space
46. Rename file
47. Sort directory by date, by file name
48. Control directory view to show file names and dates at left
49. Create folder, move files to folder, move folder
50. Create shortcut
51. View print queue
52. Create folders in your network student account area for files associated with this course and other projects. Design a file organization for your materials. For example, you may wish to create folders within the course folder for each topic, or a folder homeworks and one for the project, etc. Submit a screen shot of your student account file directory using Word or Paint.
53. Download the program *asm_setup.exe* (see below) and use it to install the program *asm.exe* in your student account (see below). Run *asm.exe*, and use it to step through the program *xy.asm* (below, and available in Blackboard under Course Documents), choosing step mode. To open *xy.asm* within *asm.exe*, use File / Open.

```
input x
load x
add x
add x
store y
print y
stop
```

Test the program for two or more different input values.
(a) From your observation, what occurs when the fifth line of the program, *store y*, executes?

(b) Based on your observation, write a formula that accurately describes the relationship between input and output of program *xy.asm*. (Your formula could be of the form “Output is $n$ larger than input,” or “Output is random,” or “Output is same as input.”)
(c) Suggest a name for the program that describes what it does better than the name *xy.asm*, and better names for the data labels $x$ and $y$.
(d) Using a text editor such as Notepad, copy into your homework file a listing of *xy.asm*, the program, and *xy.out*, the record of your test of the program. The file *xy.out* is created when *asm.exe* runs *xy.asm*. (Optional: add a comment with your answer to (b) and rename the program and variables per (c).)
54. Describe in your own words the process of executing a program at the hardware level, referring to the processor, machine language, bits, registers, RAM, and I/O.

**Short answer**
1. What is the basic unit of information storage in a file?
2. Name two operating systems.
3. What is a smaller unit of information than a byte?
4. What feature is found in all instances of information technology?
5. What keypress places the screen image on the Clipboard?

**Multiple choice**
1. In a user PC on the FSC network, the “Y: drive” is (a) a processor; (b) a physical device; (c) a folder on the user computer’s hard drive; (d) located on the server; (e) none of these
2. An example of analog representation is (a) a file stored on a computer; (b) a message sent on the Internet; (c) the sound heard from an IPod; (d) a picture in RAM; (e) a register in a processor
3. A bit’s value (a) is 0 to 255; (b) is 0 or 1; (c) fills a register; (d) fills a memory cell; (e) corresponds to a color pixel
4. An operating system provides services for (a) applications; (b) remote sites; (c) hardware; (d) Microsoft Corp.; (e) surgeons
5. Which is not hardware? (a) general-purpose computer; (b) operating system; (c) video game console; (d) printer; (e) all are hardware.

6. The two standard ways to access data from storage include sequential and (a) binary; (b) wireless; (c) random; (d) arbitrary; (e) reverse.

7. Components of a CPU include (a) RAM; (b) control unit; (c) track; (d) packet; (e) software.

8. Right-clicking on an object (a) opens or executes it; (b) deletes it; (c) selects it; (d) displays its operations and attributes; (e) none of these.

9. Of the following, the smallest is: (a) bit; (b) kilobyte; (c) megabyte; (d) byte; (e) word.

10. All data is communicated and stored by computers in what form? (a) analog; (b) digital; (c) megabyte; (d) packet; (e) other.

11. Which type of language is closest to that used by a processor? (a) query; (b) formula; (c) markup; (d) assembler; (e) transfer-protocol.

12. The two standard ways to access data from storage include sequential and (a) binary; (b) wireless; (c) random; (d) arbitrary; (e) reverse.

13. Consider the following eras: (i) Internet-connected computers; (ii) mainframe computers; (iii) locally-networked PCs. Computing has proceeded from (a) i to ii to iii; (b) ii to i to iii; (c) iii to ii to i; (d) iii to i to ii; (e) ii to iii to i.

14. The fastest-accessible of the following is: (a) RAM; (b) hard disk; (c) cache; (d) register; (e) web site.

15. What is fetched in the fetch-execute cycle? (a) instruction; (b) operand; (c) record; (d) byte; (e) file.

16. The kind of desktop display in our classroom is (a) PDF; (b) CRT; (c) LCD; (d) ABC; (e) none of these.
**Study questions on Topic 3 (Design)**

**Group work**

1. Consider the flowchart below. For each of the input pairs \((a, b)\), shown below, show the resulting output:

<table>
<thead>
<tr>
<th>Input a</th>
<th>Input b</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) 2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>(b) 1</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>(c) 4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>(d) 7</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>(e) 3</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>(f) 3</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>(g) 6</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

Informally, what does the output of this algorithm tell about the two inputs?

2. Use the table below to trace the algorithm specified in the flowchart above, for inputs \(a\) and \(i\) of

<table>
<thead>
<tr>
<th>(a)</th>
<th>(i)</th>
<th>(y)</th>
<th>output</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) 2</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(b) 4</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(c) 5</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(d) 10</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(e) 12</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(f) 1</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(g) 2</td>
<td>4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Use the table below to trace the algorithm specified in the flowchart above, for inputs \(a\) and \(i\) of

3. Show a trace of this flowchart of binary-to-decimal conversion, for inputs (a) 11011; (b) 101101; (c) 101010; (d) 011001; (e) 010110; (f) 100101; (g) 100111

4. (Challenge) Using flowchart or pseudocode, design an algorithm or interactive process that
(a) loops to accept four numbers and displays their sum
(b) accepts two numbers and displays the larger.
(c) accepts three numbers and displays the largest;
(d) four numbers;
(e) five numbers
(f) prompts for two integers and displays their quotient, using only subtraction; show an error message if the divisor is 0,
(g) accepts a number \(x\) and displays the sum of all the whole numbers from 1 to \(x\);
(h) accepts a number \(x\) and displays the product of all the whole numbers from 1 to \(x\)
(i) prompts for two integers and display their product, using only addition and subtraction operations in your calculations;
(j) prompts for a string and tells whether it contains any doubled-up characters.

**Group work**

1. Write a flowchart for an algorithm that inputs twenty numbers and displays their average.
2. Trace the flowchart below [see examples].
3. Consider the flowchart below. For each of the input pairs \((a, b)\), shown below, show the
resulting output:

<table>
<thead>
<tr>
<th>Input a</th>
<th>Input b</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>_____</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>_____</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>_____</td>
</tr>
<tr>
<td>7</td>
<td>10</td>
<td>_____</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>_____</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>_____</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>_____</td>
</tr>
</tbody>
</table>

Informally, what does the output of this algorithm tell about the two inputs?

4. Use the table below to trace the algorithm specified in the flowchart above, for inputs $a$ and $i$ of
(a) 2 and 3;
(b) 4 and 2;
(c) 5 and 1;
(d) 10 and 3;
(e) 12 and 2;
(f) 1 and 6;
(g) 2 and 4.

<table>
<thead>
<tr>
<th>$a$</th>
<th>$i$</th>
<th>$y$</th>
<th>output</th>
</tr>
</thead>
<tbody>
<tr>
<td>_____</td>
<td>_____</td>
<td>_____</td>
<td>_____</td>
</tr>
<tr>
<td>_____</td>
<td>_____</td>
<td>_____</td>
<td>_____</td>
</tr>
<tr>
<td>_____</td>
<td>_____</td>
<td>_____</td>
<td>_____</td>
</tr>
<tr>
<td>_____</td>
<td>_____</td>
<td>_____</td>
<td>_____</td>
</tr>
</tbody>
</table>

5. Show a trace of this flowchart of binary-to-decimal conversion, for inputs (a) 11011; (b) 101101; (c) 101010; (d) 011001; (e) 010110; (f) 100101; (g) 100111

6. (Challenge) Using flowchart or pseudocode, design an algorithm or interactive process that
(a) loops to accept four numbers and displays their sum
(b) accepts two numbers and displays the sum
(c) accepts three numbers and displays the largest;
(d) four numbers;
(e) five numbers
(f) prompts for two integers and displays their quotient, using only subtraction; show an error message if the divisor is 0.
(g) accepts a number $x$ and displays the sum of all the whole numbers from 1 to $x$;
(h) accepts a number $x$ and displays the product of all the whole numbers from 1 to $x$
(i) prompts for two integers and display their product, using only addition and subtraction operations in your calculations;
(j) prompts for a string and tells whether it contains any doubled-up characters.

7. What is an algorithm?
8. Distinguish problem specification, system design, and program coding.
9. Put each of the following into one of the shapes below (rectangle, parallelogram, or diamond):
(a) count < 5; (b) display total; (c) total ← total + $x$.

10. What are two ways to express an algorithm?
11. What kind of structures are the sequence, branch, and loop, and what algorithms can they be used to specify?
12. What are the three standard control structures, sufficient to specify any algorithm?
Multiple choice

1. Modular decomposition of processes is most closely associated with which kind of design?  
(a) web-site formatting; (b) spreadsheet; (c) database; (d) algorithm; (e) none of these
2. In event-driven programming, an event is (a) input; (b) output; (c) a sequence structure; (d) a program decision; (e) something that happens during web-site development
3. Which is not a feature of algorithms?  
(a) precision; (b) finiteness of time; (c) step-by-step sequencing; (d) limited set of possible inputs; (e) definiteness of result
4. Which is not a way to express an algorithm?  
(a) HTML; (b) JavaScript; (c) flowchart; (d) pseudocode; (e) machine code
5. Which of these is a control structure?  
(a) hyperlink; (b) Excel worksheet; (c) database table; (d) loop; (e) register
6. Algorithms (a) are efficient; (b) take finite time; (c) are languages; (d) are a kind of program; (e) none of these
7. Design tools include (a) output; (b) flowcharts; (c) registers; (d) queries; (e) none of these
8. The loop is a (a) language; (b) control structure; (c) data structure; (d) program; (e) none of these
9. The branch is a (a) language; (b) control structure; (c) data structure; (d) program; (e) none of these
10. Control structures are used in (a) design; (b) output; (c) input; (d) formatting; (e) none of these
11. A trace of an algorithm provides (a) input; (b) a list of errors; (c) snapshots of the state of the algorithm over time; (d) a view of a table; (e) none of these
12. One language used for design of interactive systems is (a) HTML; (b) machine language; (c) database query language; (d) UML; (e) none of these

Multiple-choice or T/F:

1. A precise plan to solve a problem in finite time is (a) a module; (b) a function; (c) an algorithm; (d) recursion; (e) none of these
2. (T-F) A recursive definition is one that has just one case to handle.
3. (T-F) At the processor level, a loop entails a backward jump.
4. Which is not a recommended tool for program design? (a) flowcharts; (b) pseudocode; (c) object-oriented analysis; (d) hierarchy charts; (e) use of keywords
9. A data item that is defined in terms of properties and operations is (a) simple; (b) a bit; (c) input; (d) an object; (e) binary
10. (T-F) A flowchart using only three different control structures can diagram a solution to any solvable problem.
11. Pseudocode (a) has a precise syntax; (b) is a false solution; (c) is a low-level language; (d) is an informal notation; (e) none of these
12. The branch is a (a) hardware item; (b) control structure; (c) data structure; (d) flowchart rectangle; (e) module
13. (T-F) The problem-solving process presented in class places design before coding.
14. (T-F) The body of a top-tested loop will always execute at least once.
15. Object-oriented design focuses problem solving on (a) categories of things; (b) processes; (c) functions; (d) integers
16. Successively more detailed development of an algorithm is called (a) object-oriented design; (b) a module hierarchy; (c) bottom-up design; (d) stepwise refinement; (e) problem specification
17. Multiple alternatives (a) are not supported by standard programming; (b) call for use of modules; (c) require multiple diamonds in a flowchart; (d) require repetition; (e) none of these
18. When a problem is complex, the complexity can often be conquered in the design stage by (a) brute force; (b) documentation; (c) modular decomposition; (d) input/output; (e) logic gates
19. (T-F) A branch should not be nested inside a loop.
20. (T-F) A component of a structured flowchart has one entrance and one exit.
21. (T-F) The loop is a data structure.
22. (T-F) Structured programming and modular programming are roughly synonyms.

Short-answer:
1. The word otherwise might be appropriate in pseudocode for which control structure?
2. In a module hierarchy, what is the location of a module used directly by module x?
3. What is an algorithm?
4. Name the three fundamental control structures recommended in structured programming.
5. In software development, what steps are recommended before coding?
6. Name four chief tools used for writing down a program design before coding.
7. What is the output of the program diagrammed by the flowchart below?

```
Begin
  qty ← 2
  qty > 9? T
    Display qty
    Add 3 to qty
  qty < 9? F
    End
```
8. What output would be produced by a program based on this flowchart, on input 12?

```
Begin
  Set product to 2
  Set log to 0
  Input argument
  argument < product Y
    Double product
    Add 1 to log
  argument ≥ product N
    Display log
End
```
9. What output would be produced by a program based on this pseudo-code, on inputs of 70 and 80?

```
Begin
  Input a, b
  result ← a
  b ≤ 0? Y
    Add 1 to result
    Subtract 1 from b
  N
    Display result
End
```
10. What kind of programming stresses use of three control structures and modular decomposition?
11. What kind of programming stresses defining categories of things and their behaviors?
12. The word repeat might be appropriate in pseudocode for which control structure?
13. Counting from 5 to 15 would require which control structure?
14. What is the computer-science term for a precise plan to solve a problem in a finite amount of time?
15. What is pseudocode?

17. In the software development process, what steps are recommended before coding a program? After coding?

18. Give an example of an object that is found on the screen in the Windows or OS/2 user environment. What are some of its data attributes? Its behaviors?

19. How many times will a counter-controlled loop iterate? A sentinel-controlled loop?

20. Name two variants of the loop control structure.

21. Consider the flowchart below. For each of the input pairs (A,B) shown below, show the resulting output.

<table>
<thead>
<tr>
<th>Input A</th>
<th>Input B</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>_____</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>_____</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>_____</td>
</tr>
</tbody>
</table>

What simple function does this algorithm calculate?

22. Put these phases or sub-phases of the problem-solving process in chronological order, number the first “1”, the second “2”, etc.

- code program
- desk check
- write a design
- get problem specifications
- test program
- debug code

23. Consider the flowchart below. For each of the input pairs (A,B), shown below, show the resulting output.

<table>
<thead>
<tr>
<th>Input A</th>
<th>Input B</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>_____</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>_____</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>_____</td>
</tr>
</tbody>
</table>

What is a way to describe the relationship between input and output?

24. Label each term below with the letter of its appropriate definition:

- pseudocode
- object
- algorithm
- branch
- loop
- module
- desk checking
- stepwise refinement
- top-down design

a) A precise plan to solve a problem or complete a task in a finite number of steps.
b) Informal natural-language way to express an algorithm.
c) The decision control structure, in which one action is taken or else another.
d) The iteration control structure, in which an action is repeated.
e) A way to design and code software characterized by use of only three control structures: sequence, branch, and loop.
f) A data item that is defined partly in terms of its behavior.
g) Verification of program correctness without running it on a computer.
h) A program component which may consist of one or more subprograms.
i) A method of developing a plan for a program, beginning with an overview of the problem and breaking it down.
j) A method that uses repeated improvements in a program design.

**Longer-answer questions:**

1. Is the pseudocode below an example of a structured design? Why or why not?
   1. If input file exists, open input file; else exit program
   2. Read input file, summing up contents
   3. Display sum.
2. Modify the flowchart below so that it will diagram a structured design.

3. Write a flowchart or pseudocode for algorithms to solve each of the following problems.
   (a) Input exactly 6 signed integers. Display only the largest of the input values, regardless of where it occurred in the input list. Hint: Let the data address for the first input also serve as the storage location for the largest integer found so far. Use a second data address for subsequent input values.
Sample I/O: \(-3,20,-4,5,7,0\) Output: 20

(b) Input signed integers until the current input is less than the previous input. Display the largest input value.
Sample Input: \(1,2,3,4,24,56,41\) Output: 56

4. Use pencil and paper to test a few argument values and determine what familiar mathematical functions are computed by the following recurrences:

(a) \(f(a,b) = \begin{cases} 0 & \text{if } a = 0 \\ b + f(a-1,b) & \text{otherwise} \end{cases} \)

(b) (Challenge) \(g(a,b) = \begin{cases} 0 & \text{if } a = 0 \\ b + g(\lfloor \frac{a+2}{2} \rfloor, 2b) & \text{if } a \text{ is odd} \\ g(\lfloor \frac{a+2}{2} \rfloor, 2b) & \text{if } a \text{ is even} \end{cases} \)

5. Design a program to accept keyboard input of three integers to represent the dollar amounts price, discount, and sales tax, in cents. It should display the sum of the price and the tax, minus the discount.

Sample I/O:
[Input]: 200
[Input]: 20
[Input]: 10
[Output]: 190

6. Computers are often sold with service plans whose cost depends on the computer’s value. Design a program to input the signed integers, price and monitor. If the sum of these is less than 1000, the program should display the message, “Plan costs $99.95”; otherwise it should display the message, “Plan costs $149.95”.

Sample I/O:
Example 1: [Input]: 749 [Input]: 399 [Output]: Plan costs $99.95
Example 2: [Input]: 1299 [Input]: 399 [Output]: Plan costs $149.95

7. Design a program to input three integers, A, B, and C. Make the necessary comparisons to display the greatest of the three.

Sample I/O:
[Input]: -38
[Input]: 300
[Input]: 77
[Output]: 300

8. Design a program to input integers, A and B. Compute and display \(|A – B|\). Note: Read \(|A – B|\) as “the absolute value of the difference (A – B)”.

9. Design a program without input that uses a loop structure to display each of the integers from 1 through 10. The only data values that may be stored initially via data statements are 0, 1, and 10.

10. Design a program that will loop to accept input of exactly three pairs of integers (A,B) and compute and display the value of A – B for each input pair.

11. Design a program to compute and display the product of two input non-negative integers. Display nothing if input includes a negative number. (Hint: Perform the multiplication as repeated addition, using one of the integers as an addend and the other integer as a counter to determine how many times to add the addend to a sum representing the product.)

12. (Challenge) Design a program to input two non-negative integers (A,B) and a positive integer (C). Compute \(A * B / C\) and display the quotient and remainder. For each of the inputs A, B, and C, loop for new input if negative values are entered. The input value of C must also be tested to be sure it is not 0. Why?

13. (Challenge) A geometric progression is a sequence of terms in which each term after the first term is obtained by multiplying the previous term by a constant multiplier. For example, if the first term is 7 and the constant multiplier is 3, then the resulting geometric progression is:

\[ 7, 21, 63, 189, 567, 1701, \ldots \]

14. We can compute the sum of the first \(n\) terms of a geometric progression. In the preceding example, the sum of the first 5 terms is

\[ 7 + 21 + 63 + 189 + 567 = 847. \]

15. Design a program that will accept positive integers \(n, first, \text{ and } k\), input by the user, and display the first \(n\) terms and then the sum of those terms where the first term is \(first\), the constant multiplier is \(k\), and the desired number of terms is \(n\).

16. (Challenge) Design a program to divide any signed integer by any other non-zero integer, using only addition, subtraction, and the three control structures. First the dividend and then the divisor are to be input from the keyboard. The divisor must be tested to avoid attempted division by 0, since division by 0 is not defined. The result is to be output as an integer quotient followed by an integer remainder. Remember to test your program with all possible sign combinations of the dividend and divisor.

17. Write pseudocode to compute the base-2 integer logarithm of an input integer. (See flowchart below.)

18. Write an algorithm to find the tallest person in a room by comparing two persons at a time. \(Hint: some persons will only have to be compared only once to any other person.\) Once you compare the shortest person with even one other person, for example, you will know enough never to compare that short person again with anyone.

19. Write pseudocode or a flowchart for a program that displays each number from 1 to 100, putting an asterisk before each number that ends in ‘7’.
20. Write pseudocode or a flowchart for a program that repeatedly prompts for two quantities, displaying the larger one each time, and terminating when both quantities have the same value.

**Answers to study questions for topic 3**

**Multiple-choice or T/F:**

1. c. An algorithm must be specific, detailed, and not take forever.
2. f. A recursive definition, like "My ancestors are my parents and their ancestors," involves a base case ("my parents") and a recursive case ("their ancestors").
3. t. To repeat an instruction the microprocessor must jump to an earlier program instruction.
4. e. Stream output is a tool in coding Java programs.
5. d. The bubble sort is a full algorithm, composed of several control structures.
6. t. Iteration, or repetition, is the loop control structure.
7. b. A jump forward implements the branch control structure.
8. f. Program design is independent of languages and their grammar rules.
9. d. Object-oriented design looks at things that we deal with and how they behave.
10. t. The three control structures of structured programming are the sequence, the branch, and the loop.
11. d. Pseudocode is considered useful in program design.
12. b. The branch is the selection control structure.
13. t. Program coding is based on a thought-out design.
14. f. If the test fails the first time, the body will not execute.

15. a. An object is defined by attributes and behaviors.
16. d. The design may be refined or improved step by step.
17. c. A multiway branch corresponds to a series of cascading flowchart diamonds.
18. c. Modular decomposition is a divide-and-conquer strategy for problem solving.
19. f. Any control structure may be nested inside another.
20. t. There should be one way into and one way out of any part of an algorithm.
21. f. The loop is a control structure.

**Short-answer:**

1. branch
2. A module used by module x will be just below x in a module hierarchy chart.
3. A precise plan to solve a problem in a finite number of steps
4. Sequence, branch, loop
5. Problem analysis or specification; design
6. Pseudocode; flowcharts; module hierarchy diagrams
7. 2 5 8
8. 54321 54321
9. Tall
10. structured programming
11. object-oriented programming
12. loop
13. loop
14. algorithm
Study questions on Topic 4: Java basics

1. Write a program, with documentation to:

2. Display the words “This is a program” on four lines

3. Initialize and display four integer variables, with values 1, 2, 3, 4

4. As #2, but label output with variable names

5. Input and display four integers

6. Display
   
   \[ x \]
   \[ xx \]
   \[ xxx \]
   
in one statement.

7. Declare named constants for tax as a percentage (5) and discount (33). Display amount due on 4 inputs, consisting of a price and quantity for each of two items. Use no fractions, use integer division if necessary.

8. Multiple-choice or T/F:

   1. (T-F) The compiler ignores white space
   2. (T-F) Java has two kinds of statement: declarations and executable statements.
   3. (T-F) Java is higher-level than assembler language.
   4. (T-F) Semantics deals with structure, syntax with meaning;
   5. The \#include statement (a) treats a file as if it were part of the source file to be compiled; (b) opens a file for input; (c) opens a file for output; (d) is required for all type declarations
   6. (T-F) A library is a kind of program documentation.
   7. << is (a) an input operator; (b) an output operator; (c) the assignment operator; (d) a relational operator
   8. (T-F) Java programs use streams to handle input and output.
   9. \textit{cout} is (a) a function; (b) a keyword; (c) a data item; (d) stream input; (e) none of these
   10. A runtime package (a) is an add-on; (b) stays in the compiler; (c) is used mostly in debugging; (d) is part of every compiled program; (e) arrives in the mail of good programmers
   11. (T-F) A machine-language instruction usually performs a more complex task than a Java statement.

12. The compiler produces (a) high-level code; (b) machine code; (c) documentation; (d) keyboard input

13. Java syntax requires that every program have: (a) a \textit{main} function; (b) a variable; (c) input; (d) output; (e) comments

14. How many reserved words does Java have? (a) none; (b) a few; (c) dozens; (d) hundreds; (e) thousands

15. (T-F) Java is an extension of C.

16. (T-F) Any Java program is nearly certain to compile on a C compiler.

17. (T-F) The compiler translates from machine language to Java.

18. (T-F) \{\} is a valid statement in Java.

19. A variable does not have a (a) name; (b) address; (c) type; (d) precedence

20. To document our source code we are encouraged to use (a) short variable names; (b) comments; (c) operators; (d) manuals

21. A global data item that might be of value in documenting code is a (a) named constant; (b) comment; (c) literal; (d) variable

22. Syntax is (a) documentation; (b) meaning; (c) grammar rules; (d) good-programming guidelines; (e) recursion

23. The C or Java statement \( x = 2; \) corresponds to the processor-level operations (a) load 2 from memory, then store to location \( x \); (b) store at \( x \), then load from memory; (c) unconditional jump; (d) conditional jump backward; (e) conditional jump forward

24. (T-F) A variable is a memory location with a name and a data type.

25. (T-F) Identifiers usually stand for memory locations.
31. Variable declarations in a list following a type name are separated by (a) periods; (b) commas; (c) semicolons; (d) colons; (e) parentheses.
32. (T-F) The same kind of program elements can be on the left and right side of "=".
33. `const int x = 3;` is (a) a syntax error; (b) an assignment statement; (c) the declaration of a named constant; (d) an output statement.
34. For input in Java, we use (a) the assignment operator; (b) data statements; (c) `read();` (d) `cin;` (e) `cout`.
35. Which is the most important, for a program to be maintainable? (a) code has no syntax errors; (b) code generates no runtime errors; (c) code runs fast; (d) code is easy to understand; (e) none of these is important.
36. (T-F) The compiler reads machine code and outputs high-level code.
37. (T-F) A data type is a memory location with space allocated for a value.
38. Which determines whether a program compiles: (a) code is easy to understand; (b) code has no syntax errors; (c) code runs fast; (d) code generates no runtime errors; (e) none of these is important.
39. Which is not a valid C/Java identifier: (a) `2days;` (b) `myvar;` (c) `Data12;` (d) `FooBar;` (e) `x`.
40. (T-F) I/O takes the form of streams.
41. Write a C or Java statement that declares an integer variable, `sum`, and initializes it with the sum of `x` and 2.
42. Write a Java statement that prompts for an integer variable, `quantity`, with appropriate labelling.
43. What may the programmer do in a C or Java program to make sure his or her intention is clear?
44. Name three ways to give a value to a variable in C or Java.
45. Write a C or Java-language statement that displays the label “The quantity is” and the value of the integer variable, `quantity`.
46. Give two reasons for using named constants in a program.
47. How does a variable differ from a literal?
48. How is Java like English? How is it different?
49. What should every program have that the compiler will ignore?
50. Write a Java statement to prompt the user for a quantity of items.
51. Name a Java grammar rule that is recursive.
52. Write a C/Java statement that declares a variable that will store the number of people in a team.
53. What is wrong with this definition of a named constant?
54. Write a C or Java statement that displays the label “The quantity is” and the value of the integer variable, `quantity`.
55. Which is risky about this code?
56. What is the output of this code?
57. Write three Java statements to declare, prompt for, and input an integer variable that expresses the number of years in the term of a contract.
58. Name two different forms of documentation that you can write in your source code to make the program easier to understand.
59. Which of the items below are valid C or Java identifiers? (a) `A12;` (b) `cat;` (c) `39G;` (d) `out.put;` (e) `A3D;` (f) `employee-salary;` (g) `employee_salary`.
60. Will these two lines of source code compile? If not, what message is generated? Correct the error.
61. Declare a named constant to represent the number of days in a week.
62. Correct the error in this code.
   ```
   int quantity,price;
   cout << "Enter quantity " << quantity << " and price: ";
   cin >> quantity, price;
   cout << "Amount due: ";
   cout << quantity * price << endl;
   ```
   (Sample I/O (incorrect; fix it to output 40):
   Enter quantity and price: 10.4
   Amount due: 42890520
   )
63. Label each term below with the letter of its appropriate definition:
   ____ assignment statement
   ____ compiler
   ____ compound statement

---

Short-answer:

1. Write a C or Java statement that declares an integer variable, `sum`, and initializes it with the sum of `x` and 2.
2. Write a Java statement that prompts for an integer variable, `quantity`, with appropriate labelling.
3. What may the programmer do in a C or Java program to make sure his or her intention is clear?
4. Name three ways to give a value to a variable in C or Java.
5. Write a C or Java-language statement that displays the label “The quantity is” and the value of the integer variable, `quantity`.
6. Give two reasons for using named constants in a program.
1. Write a well-documented C or Java program that prompts for a unit price of an item and a quantity, and displays the total price to be charged.

2. Write a Java program that accepts three weights and displays their average.

3. Write a well-documented C or Java program that prompts for three quantities and displays their sum.

4. Write a Java program that declares a named constant to represent the number of weeks in a year, prompts for and inputs the number of weeks a person will go on vacation, and displays the number of work weeks in the year. (There are 52 weeks in a year.)

5. Consider the program:

```cpp
// add.cpp
// Displays sum of 2 input integers.
#include <iostream.h>
void main()
{
    cout << "Enter 2 #s to add: ";
    int input1,input2,sum;
    cin >> input1;
    cin >> input2;
    sum = input1 + input2;
    cout << input1 << " + " << input2 << " = " << sum;
}
```

If, in running this program, 20000 is input for A and 30000 is input for B, what output is generated by the last statement? Can you explain how this output value was generated? Does the discussion of integer storage and twos complements in Chapter 1 offer any hints?

6. Modify program `hello.cpp`, below, so that the output of the `cout` statement goes to a text file called `hello.out` on your work disk.

```cpp
void main()
{
    println "Hello";
}
```

7. Write, compile, and run a Java program that calculates and shows the integer $y$ for any integer input value $x$, where $y = 3x^2 - 2x + 5$. The program should send output to a disk file.

8. Will the two-line statement:
```cpp
cout << "This continues on a second line";
``` 
compile? If not, what message is generated? Correct the error and explain the general rule that applies.

9. Debug this code:
```cpp
int quantity,price;
cout << "Enter quantity " << "and price: ";
cin >> quantity, price;
cout << "Amount due: " << quantity * price << endl;
```

10. Write a well-documented Java program that declares a named constant to represent the number of days in October, prompts for and inputs the number of weekend days that month, and the number of holidays, and displays the number of remaining, work days in the month.
answers to study questions on Intro to Java

Multiple-choice or T/F:
1. t. The use of spaces, tab characters and newline characters generally has no effect on program compilation.
2. t. Executable statements include assignments and function calls; declarations include variable declarations and type definitions.
3. t. The vocabulary of Java contains powerful statements like if and while not duplicated in assembler language.
4. f. Semantics deals with meaning, syntax with structure.
5. a. The #include causes the named file to be read and compiled with the rest of the file to be compiled.
6. f. A library is a coding resource, often precompiled.
7. b. The insertion operator << is used with cout and other output streams to insert characters into the data stream.
8. t. Examples: cout, cin.
9. c. cout is an instance of an output stream type.
10. d. All low-level input and output routines, for example, are put into the executable program file by the compiler as part of a runtime package.
11. f. A C or Java assignment, for example, executes at least a load and a store instruction.
12. b. A compiler translates from a language like Java to an assembler or machine language of a particular machine.
13. a. Variables, input, and output are nearly universal but optional. The function called main is mandatory.
14. c. Java as defined by Stroustrup has 48 reserved words.
15. t. Java contains all the syntax of some additional rules.
16. f. The converse is true, but many Java programs contain statements that a C compiler would reject, or even comments, such as those beginning "//".
17. f. The compiler translates from the higher-level language to machine language.
18. t. The braces enclose a compound statement; the ones above enclose the null statement.
19. d. Variables have identifiers, memory locations, and data types.
20. b. Comments should be used to clarify a programmer’s intention.
21. a. A named constant documents itself.
22. c. Syntax rules specify the proper order of language elements in a program.
23. a. The assignment statement copies a value but leaves it intact at its source.
24. t. A variable is a cell or series of cells that may be assigned a value. To know how much space to assign the variable, the compiler must know its data type.
25. f. The Java compiler is case sensitive.
26. f. An identifier must start with a letter or an underscore.
27. c. An identifier may not begin with a digit.
28. b. An identifier may not contain a space.
29. b. Executable machine code is in binary form.
30. t. For example, a variable is a name for a data location in RAM.
31. b. Example: int a,b,sum;
32. f. Any expression may be on the right, but only variable names may be on the left.
33. c. A named constant is the declaration and initialization of a variable whose value may not be changed.
34. d. The stream object, cin, permits input from the keyboard; cout permits screen output.
35. d. For a program to be maintainable, to fix errors and add features, the programmer must be able to understand it.
36. f. The compiler reads high-level code and outputs machine code.
37. f. A data type specifies the meaning of a variable; a variable is a data item with a memory location and a value.
38. b. The compiler will be unable to compile a program that has syntax errors.

Short-answer:
1. int sum = x + 5;
2. cout << “Enter quantity: “; cin >> quantity;
3. Write comments or use meaningful identifiers
4. 1. Initialize in declaration
   2. Assign with assignment operator (=)
   3. Input with cin or scanf
5. `cout << "The quantity is " << quantity << endl;
    printf("The quantity is \%d\n", quantity);

6. (1) To document meaning of a value used; (2) To permit updating in one step a constant used multiple times

7. A variable has an address and its value can change.

8. *Like*: Has grammar rules and vocabulary

   *Different*: English has more rules, is ambiguous

9. Comments, white space

10. `Int quantity;
    cout << "Quantity: "; cin >> quantity;

11.
12.
13.
14.
15.
16.
17. `cout << "Number of years: ";
    int num_years;
    cin >> num_years;`
Study questions on Numeric data

1. Write a program that prompts for a unit price, discount rate, and quantity and displays a subtotal without tax; a tax amount, given a constant tax rate of 5%; and a total amount due. Format to two places.

2. Write a program that prompts for the height and diameter of an oil drum and displays the area of its surface, given that the area of a circle is \((p \cdot radius^2)\) and \(p = 3.14159\). Format to two places.

Multiple-choice or T/F:

1. (T-F) The compiler performs multiplication before addition in the absence of parentheses.

2. * is (a) a sign; (b) a string; (c) a function; (d) an operator; (e) a call

3. In an assignment like \(a = x\), data is (a) just copied; (b) copied and deleted; (c) discarded; (d) crunched; (e) displayed

4. Before outputting a variable, it should always be (a) assigned a value; (b) input; (c) processed; (d) evaluated

5. Before use in a statement, a variable must be (a) assigned a value; (b) declared; (c) output; (d) used in an expression

6. \(1+2*4 = (a) 0; (b) 1; (c) 8; (d) 9; (e) 12\)

7. In Java, \(15/12 = (a) 0; (b) 1; (c) 1.25; (d) 12; (e) 15\)

8. \(15 \% 12 = (a) 0; (b) 1; (c) 2; (d) 3; (e) 12\)

9. \(1998 \% 100 = (a) 19; (b) 19.98; (c) 98; (d) 199800\)

10. Convert to C or Java:

\[
\frac{b + c}{d + e}
\]

(a) \(a\); (b) \(abcde\); (c) \(b+c/(d+e)\); (d) \(b+c/d+e\); (e) \(b/d+c/e\)

11. (T-F) An item of type int occupies fewer bits than a char item.

12. A category that defines the meaning or interpretation of a pattern of bits is (a) an algorithm; (b) a register; (c) a function; (d) a data type; (e) a library

13. \% is the _______ operator. (a) int; (b) initialization; (c) modulo; (d) division; (e) insertion

14. (T-F) Unary operators have two operands.

15. Which operator could help us shorten a statement like \(total = total + price\)? (a) \(+\); (b) \(=\); (c) \(+\); (d) \(+=\); (e) 

16. Which operator could help shorten a statement like \(total = total + 1\)? (a) \(+\); (b) \(=\); (c) \(+\); (d) \( *=\); (e) 

17. If \(n\) equals 6, then \((++n)\) equals (a) 5; (b) 6; (c) 7; (d) 12; (e) 1

18. If \(n\) equals 3, then \((n++)\) equals (a) 1; (b) 2; (c) 3; (d) 4; (e) 6

19. What is the smallest capacity data type, of the following? (a) short int; (b) int; (c) unsigned int; (d) long int; (e) none of these

20. What is the largest capacity data type, of the following? (a) short int; (b) int; (c) unsigned int; (d) long int; (e) none of these

21. Which type stores its data in three components? (a) int; (b) double; (c) short; (d) char; (e) long

22. (T-F) A data item of type int can be positive or negative.

23. (T-F) A data item of type unsigned can be positive or negative.

24. (T-F) A value of type float can be negative.

25. (T-F) A value of type double can be between 0 and 1.

26. (T-F) A value of type int can be between 0 and 1.

27. (T-F) Floating-point storage entails representational error.

28. (T-F) Since \((6 - 3 - 1)\) is 2 rather than 4, therefore we know that the subtraction operator is right associative.

29. A mantissa, or fraction, component appears in type (a) double; (b) int; (c) char; (d) unsigned; (e) string

30. (T-F) Floating-point storage uses an organization concept similar to scientific notation.

31. What is necessarily double about the type double? (a) magnitude; (b) storage; (c) precision; (d) meaning; (e) sign

32. Standard functions abs, sin, exp, and sqrt are defined in the library file (a) iostream.h; (b) ctype.h; (c) stdio.h; (d) math.h; (e) stdlib.h

33. Type casting in C/Java (a) is automatic; (b) is a way to produce a fractional value when dividing two integers; (c) uses the keyword throw; (d) is discouraged; (e) is a syntax error

34. To format output, we can use (a) a math library; (b) iostream.h; (c) manipulators; (d) cin; (e) >>

35. An item of type char occupies how many bits? (a) 0; (b) 1; (c) 4; (d) 8; (e) 16

36. With what special tool can you explicitly specify the width of part of your output? (a) cout; (b) cin; (c) setw(); (d) ios; (e) setiosflags()

37. What is the data type of 3.14? (a) char; (b) int; (c) double; (d) char[7]; (e) void

38. What is the data type of 2.0? (a) char; (b) int; (c) double; (d) char[7]; (e) void
39. To convert an int value to a float in Java (a) is possible with just an assignment; (b) requires function calls; (c) requires writing an algorithm; (d) is not possible

40. If PI is 3.1416, and n is declared as an integer, then 
\[ n = 2 * PI; \text{cout} \ll n; \] yields (a) 0; (b) 3; (c) 6; (d) 6.2832; (e) a compiler error

41. (T-F) If \( x \) is an integer variable and \( y \) is a double, then 
\[ x = y; \] will compile.

42. (a) ; (b) ; (c) ; (d) ; (e)

43. (a) ; (b) ; (c) ; (d) ; (e)

44. (a) ; (b) ; (c) ; (d) ; (e)

45. (a) ; (b) ; (c) ; (d) ; (e)

Short-answer:
1. Evaluate the C/Java expression, \( 1 + 7 * 4 + 15 / 3 \).
2. What is the output of this code?
```cpp
int n = 25.9;
cout << n << endl;
```
3. Write a statement in Java that will declare and assign the value 70.4 to a variable named, `height`.
4. What is the relationship between a data type and a variable?
5. Write an expression that converts the value 8.42 to int and stores the result in an int variable, \( n \).
6. Write a declaration for a variable, \( cost \), that can store a fractional value.
7. Name two data types whose instances can store fractional values.
8. Name two data types whose instances can store whole-number values.
9. List the arithmetic operators in order of precedence.
10. Write an expression whose value is the remainder when 7 is divided by 2.
11. What header file is used to access the functions `abs`, `sqrt` and `sin`?
12. Write a C/Java expression equivalent to \( 5x^3 \)
13. Write a shorter version of this statement: `quantity = quantity + input;`
14. What operator assigns to its left operand the product of its left and right operands?
15. What operator assigns to its left operand the sum of its left and right operands?
16. What operator assigns to its left operand the difference between its left and right operands?
17. What operator assigns to its left operand the quotient of its left and right operands?
18. What operator assigns to its left operand the remainder when its left operand is divided by its right?

19. Write an expression that increments a variable \( n \) by 1 and whose value is the original value of the variable.
20. Write an expression that decrements a variable \( n \) by 1 and whose value is the original value of the variable.
21. Write an expression that increments a variable \( n \) by 1 and whose value is the incremented value.
22. Write an expression that decrements a variable \( n \) by 1 and whose value is the decremented value.
23. Name an integer data type that may store a larger range of values than an int.
24. Name an integer data type that may store a smaller range of values than an int.
25. Name a numeric data type that stores only values greater than or equal to 0.
26. Evaluate: \( 2 - 6 / 2 + 4 \)
27. What standard library defines the `sqrt` function?
28. What is the common term for coercion of data types?
29. Write an expression that type casts the double variable `amt` to type int.
30. When a value assigned to a variable is too large for the variable’s storage capacity, it is a case of __________.
31. What is the output of:
   ```cpp
   int a = 12.8;
cout << a;
   ```
32. What is the output of:
   ```cpp
   int a = 9.6;
cout << a;
   ```
33. What is the standard library used in formatting numeric output?
34. List from smaller to larger (in storage space occupied) the following types: int; float; short; unsigned; double; char.
35. Write an appropriate constant declaration allowing you to replace the numeral 0.05 in the following Java statement with a meaningful identifier.
   ```java
   tax = 0.05 * sales_amt;
   ```
36. Will the following program compile? If not, what error message will you get?
```cpp
void main()
{
    const int num_bldgs = 4;
    num_bldg = 5;
}
```
38. How many different floating-point numbers can be represented if 16 bits are allocated for an exponent and 31 bits are allocated for the normalized binary fraction? What range of values can be represented?

39. These declarations will be used to compute the area of a circle from an input radius value:
   ```java
   float a, r;
   ```

40. Rewrite the declarations, using meaningful identifiers.

41. Is a real number a kind of rational number? Explain with reference to the Venn diagram in Section 3.2.

42. Why is it impossible to exactly represent the value (1/3) in binary floating-point notation? Why is it impossible to represent every square root exactly?

43. What is the risk of using `cin` to input data to `int` or `float` variables?

44. What is the difference among the Java constants '5', "5", 5, and 5.0?

45. Write a `cout` statement that displays the `float` variable `length`, of value 2.54, as "2.5".

46. Write a `cout` statement that displays the total price of 127 items, at $0.326 apiece, rounded to the nearest cent.

47. What is wrong with these named-constant definitions?
   (a) `const float LB_PER_OZ = 1/16;`
   (b) `const TAX_RATE = 0.05;`

48. What is the value of `pounds` after these two lines execute? (Careful…)
   ```java
   int ounces = 8;
   float pounds = ounces / 16;
   ```

49. Using the C/Java precedence rules, evaluate each of the following expressions:
   (a) `12 / 4 – 2` ________
   (b) `3 + 4 * 5` ________
   (c) `7 % 3 + 1` ________
   (d) `20 / (4 + 3 * –2) * 3` ________

50. Problem: Several people share the cost of a computer list-priced at $599.95; it is sold at a 10% discount and has a 5% sales tax.
   (a) Write a C or Java expression that represents the price, in dollars and fractions of a dollar, of one person’s share of a computer if it retails at $599.95, and is shared among the members of a club that has five members now and will have two more members when the computer is purchased.
   (b) Write an expression using variables for price, discount, etc.

51. Write a C or Java statement that calculates and displays the corresponding integer `y`, for any integer input value `x`, applying the formula:
   ```latex
   y = \frac{3x^2 - 2x + 5}{4}
   ```

52. What are the outputs of each of these statements?
   (a) `cout << 1 / 3;`
   (b) `cout << (8 / 2) / 3;`
   (c) `cout << (8 / 2.0) / 3;`
   (d) `cout << 26 % 4;`
   (e) `cout << 7.5 / 2.5;`
   (f) `cout << 2.5 * 4;`

53. Write an appropriate constant declaration that would allow you to replace the numeral 0.05 (for sales tax) in the following Java statement with a meaningful identifier.
   ```java
   tax_amt = 0.05 * sales_amt;
   ```

54. Using type casting, write a statement that assigns the value 2/3 (the mathematical quotient of 2 and 3) to a `double` variable.

55. Label each term below with the letter of its appropriate definition:
<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>long</code></td>
<td>A format for storing possibly fractional numbers</td>
</tr>
<tr>
<td><code>floating-point storage</code></td>
<td>Discrepancy between a floating-point numeral as stored, and the intended value.</td>
</tr>
<tr>
<td><code>representation error</code></td>
<td>A way to give a value to a data item in the same statement that declares it.</td>
</tr>
<tr>
<td><code>initialization</code></td>
<td>Data type for possibly fractional values.</td>
</tr>
<tr>
<td><code>endl</code></td>
<td>Automatic feature in Java that allows assigning value of one type to variable of another.</td>
</tr>
<tr>
<td><code>float</code></td>
<td>Standard constant that generates new line of output.</td>
</tr>
<tr>
<td><code>type conversion</code></td>
<td>Standard library file that supplies output-formatting functions.</td>
</tr>
<tr>
<td><code>iomanip.h</code></td>
<td>Standard function that determines field width for output.</td>
</tr>
<tr>
<td><code>setw</code></td>
<td>Any of a set of standard functions that helps format floating-point output.</td>
</tr>
</tbody>
</table>

**Longer answer problems:**

1. Write a C or Java program that prompts for the length of piece of lumber, in feet, and displays that length in meters. A meter is 39.37 inches and a foot is 12 inches.
2. Write a program to input a salesperson’s monthly sales total and base monthly salary, and compute her or his gross pay according to the formula that gross pay is the base salary plus five percent of monthly sales. Compute net pay according to the formula that net pay is eighty percent of gross. Print a monthly sales report for the salesperson as a table, using the following input/output guidelines, rounding off to the nearest cent.

Sample Input/Output:
Enter monthly sales total: 10000
Enter base monthly salary: 2000.00
Sales  Base  salary  Gross  pay  Net  pay
10000.00  2000.00  2500.00  2000.00

3. Write a program to compute the length of the diagonal of a square, prompting for the length of the edge of the square as input. The diagonal is the square root of the sum of the squares of two edges. Output a table, formatting numeric values to three decimal places.

Sample Input/Output:
Enter edge: 1.0
Edge   Diagonal
   1.000  1.414

4. Write a program to input a salesperson’s employee ID, his or her monthly sales total (dollars and cents), and his or her base monthly salary (dollars and cents). Compute gross pay = base monthly salary + 5% of monthly sales total
Compute net pay = 80% of gross pay. Display a monthly salary report for the salesperson, using the following input/output format:
Enter salesperson’s ID: 8192
Enter monthly sales total: 5560.75
Enter base monthly salary: 1500.00
Sales  Base  salary  Gross  pay  Net  pay
8192  5560.75  1500.00  1778.04  1422.43

5. Write a program to compute and print the y-coordinate (y) of a point on the straight line defined by the formula \( y = mx + b \), where \( m \) (slope), \( b \) (vertical offset), and \( x \) (the x-coordinate of the same point) are float values input from the keyboard. Format the computed value of \( y \) correct to two decimal places.

Sample I/O:
Enter M: 3
Enter: 4
Enter X: 2.75

6. Write a program that computes and prints the average speed (miles per hour) and the gas mileage (miles per gallon) for an automobile trip. The distance (miles), time (of trip in hours), and gallons (of gas used) are floating-point numbers input from the keyboard. Display computed output correct to 1 decimal place in a display field of 10 columns.

Sample I/O:
Distance: 200
Gallons: 10.5
Time: 4
Average speed: 50.0 MPH
Gas mileage: 19.0 MPG

7. Write a program to compute the maximum trip distance (in miles) possible in a car that has a gas tank capacity (in gallons) and which averages MPG miles per gallon of gas on trips. Capacity and MPG are float data input from the keyboard. Display the computed distance truncated to an integer number of miles.

Sample I/O:
Average miles per gallon: 23.7
Tank size (gallons): 15.0
Maximum trip distance without refueling: 355 miles

8. Write a program that accepts input of an integer invoice number, an integer quantity, and a real number unit price; computes a total price (quantity multiplied by unit price); and displays a simple invoice as shown below.

Sample Screen I/O:
Enter invoice number: 23001
Enter quantity ordered: 53
Enter unit price: 27.95
Invoice # 23001
QuantityUnit priceTotal price
53 27.95 1481.35

9. Write a program to compute the area of a circle if the radius is input from the keyboard. Use the \( \pi \) constant. The relationship between the radius and area of a circle is area = \( \pi \times radius^2 \). Display the area correct to 1 decimal place.

Sample I/O:
Enter radius: 10.2
Area: 326.9

10. Write a program that computes the radius of the circle whose area is input from the keyboard. Use the \( \pi \) constant and \( \sqrt{\cdot} \) function. Display the radius correct to 2 decimal places.

11. Write a program that converts an input integer number of pounds plus an input integer number of ounces into a float number of kilograms. There are 16 ounces in a pound. One pound = 0.453592 kilogram. Display kilograms correct to 2 decimal places.
12. Write a program to compute the length of the hypotenuse of a right triangle with two 45-degree angles, if the length of one of the equal legs is input. Use the \( \sin \) function for one of the equal angles; the \( \sin \) function returns the ratio of the length of the leg opposite the angle to the length of the hypotenuse. The (angle) argument for the \( \sin \) function must be expressed in radians. One degree equals (Pi / 180) radians.

13. Find the base-2 logarithm of any input value, using the \( \log \) function and the fact that for any value \( A, \log_B(n) = \log_B(A) \times \log_A(n) \). It may help you to know that the Java \( \log \) function returns \( \log_e \) of its argument, where \( e \) is approximately 2.71828.

14. Write a program that tells how many different passwords can be formed using eight uppercase letters of the alphabet. Assume that letters may be repeated, so that, for example, “reindeer” is a valid password. Do you trust the computed result? Do the assignments

\[
P = 26 \times 26 \times 26 \times 26 \times 26 \times 26 \times 26 \times 26
\]

and

\[
P = 26.0 \times 26.0 \times 26.0 \times 26.0 \times 26.0 \times 26.0 \times 26.0 \times 26.0
\]

where \( P \) is a declared float variable, give the same result? If not, which is correct? Can you explain?

15. Write a program that prompts for three locations on a set of coordinate axes. Each point should have an \( x \) and a \( y \) value. Assume that the three points form a rectangle and display the lengths of its three sides. Use a comment or screen message to document the constraints on input data needed to avoid misleading output, such as a side of length 0.

16. Display these values, accurate to three decimal places:
   (a) the square root of two; (b) \( \pi \); (c) the sine of a 45 degree angle; (d) the cosine of a 60 degree angle; (e) \( e \), the base of the natural logarithm

17. Debug this code, whose output is 0 on input of 2 lb., 3 oz.:

```java
const KILOS_PER_OUNCE = 0.0283495;
cout << "Enter pounds " "and ounces: ";
cin >> pounds >> ounces;
float total_ounces = 16 * pounds + ounces;
float kilos = total_ounces * KILOS_PER_OUNCE;
cout << "Kilograms: " << kilos << endl;
```

18. Show, through compiler error messages or successful compilations, what the syntax rules of Java are regarding the order in which the type qualifiers \( \text{const, unsigned, short, long, and double} \) may appear with the types \( \text{int} \) and \( \text{float} \).

19. Write a Java program that finds the area of a square whose side is twice the height of the user, in inches. Prompt the user for that height.

20. Write a program that displays the hour it will be, \( \text{duration} \) hours after \( \text{start} \) o’clock, given input of both integers. The program can be written using only three statements and only those Java elements presented in this chapter. On input of 11 and 1, output should be 12; on input of 6 and 7, output should be 1; on input of 2 and 24, output should be 2; on input of 12 and 120, output should be 12.

21. What is the last (rightmost) digit in the binary numeral expressing 210001? (This may be done without a program.) Write a program to output the rightmost decimal digit of the number 210001. The following table may help you:

<table>
<thead>
<tr>
<th>2^n</th>
<th>1 2 3 4 5 6 7 8</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1 2 4 8 16 32 64 128</td>
</tr>
</tbody>
</table>

22. Write a C or Java program that prompts for the costs of two items for sale, and displays the difference, accurate to two decimal digits.

**Sample Input/Output:**

```
Enter two costs: 399.95 240
399.95 - 240.00 = 159.95
```
Answers to study questions on numeric data

Multiple-choice or T/F:
1. t. The order of precedence of arithmetic operators is: unary negation, multiplication/division, addition/subtraction.
2. d. Among other things, * is the multiplication operator.
3. a. An assignment loads data into a register, then copies it to a named memory location.
4. a. Outputting an unassigned variable will yield unpredictable results.
5. b. Syntax requires declaration of all variables, but no other.
6. d. The multiplication operator is applied before the addition operator.
7. b. Integer division yields an integer quotient.
8. d. The modulo operator yields the remainder in the division of the first operand by the second.
9. c. The modulo operator yields the remainder in the division of the first operand by the second.
10. c. Parentheses are necessary to override priority of operations given to division over addition
11. f. A char occupies 8 bits (one byte); an int is usually 16 or 32 bits.
12. d. A variable is an instance of a data type; the data type is a category or description.
13. c. The modulo operator % produces the remainder of an integer division.
14. f. Unary operators are one operand operators.
15. d. The extended assignment operator for addition, +=, adds a value and produces a value.
16. c. The post or pre-increment operator, ++, adds 1 to a variable.
17. c. An expression formed with the pre-increment operator is evaluated after the incrementing.
18. c. An expression formed with the post-increment operator is evaluated before the incrementing.
19. a. Type short never takes more storage than type int.
20. d. Type long never takes less storage than type int.
21. b. Floating-point data is stored in a format with a sign bit, a mantissa (fraction), and an exponent.
22. t. The type int is signed.
23. f. An unsigned integer is always greater than or equal to 0.
24. t. Both floating-point types, float and double, have a sign bit.
25. t. Fractional values are managed by float and double.
26. f. Type int stores whole numbers only.
27. t. For values like 1/3, round-off is necessary in floating-point storage.
28. f. Left associative operators are applied from left to right.
29. a. Floating-point storage includes a binary fraction, which is multiplied by two raised to the power of the exponent component of the stored value.
30. t. Scientific notation, like floating-point storage, uses a fraction and an exponent.
31. c. The type double is double-precision floating point; that is, the mantissa may have twice as many bits as a float’s mantissa.
32. d. The C/Java mathematics library is in header file math.h.
33. b. Since integer division produces an integer result, it is necessary to type cast an operand to a floating-point type in order to produce a fraction, as in (float)2 / 3.
34. c. The manipulators defined in iomanip.h include setprecision, setw, and setiosflags.
35. d. char takes up one byte and may have any of 256 different values.
36. c. The setw stream function accepts a field width as a parameter and may be used to the left of a data item in a stream output statement.
37. c. A numeral with a decimal point is considered to be a floating-point value.
38. c. A numeral with a decimal point is a floating-point value in Java, even if it happens also to be an integer mathematically.
39. a. For example, a floating-point number assigned to an integer is truncated by the compiler.
40. c. A real number assigned to an integer variable is converted to int.
41. t. Numeric type conversion is automatic in Java.

Short-answer:
1. \( 1 + (7 * 4) + (15 / 3) = 1 + 28 + 5 = 34 \)
2. \( 25 \)
3. `scanf("%f", &height); cin >> height;`
4. A variable is an instance of a data type.
5. `int n = 8.42;`
6. `float cost`
7. `float; double`
8. `int; char; long int; short int`
9. `/`, `%`, `*`; `+`, `-`
10. `7 % 2`
11. `math.h`
12. `5 * x * x * x`
13. `quantity += input;`
14. `*=` assigns to its left operand the product of its left and right operands?
15. `+=` assigns to its left operand the sum of its left and right operands?
16. \( -= \) assigns to its left operand the difference between its left and right operands?
17. \( /= \) assigns to its left operand the quotient of its left and right operands?
18. \( %= \) assigns to its left operand the remainder when its left operand is divided by its right?
19. \( n++ \) increments \( n \) by 1 and its value is the original value of \( n \).
20. \( n-- \) decrements \( n \) by 1 and its value is the original value of \( n \).

21. \( ++n \) increments \( n \) by 1 and its value is the incremented value.
22. \( --n \) decrements \( n \) by 1 and its value is the decremented value.
23. Type \( long \) may store a larger range of values than an \( int \).
24. Type \( short \) may store a smaller range of values than an \( int \).
25. \( unsigned int \) stores only values greater than or equal to 0.
26. \( 2 - 6 / 2 + 4 = 3 \)
27. The standard library \( math.h \) defines the \( sqrt \) function.
28. Type casting is the common term for coercion of data types?
29. \( (int)\text{amt} \) type casts the \( double \) variable \( \text{amt} \) to type \( int \).
30. When a value assigned to a variable is too large for the variable’s storage capacity, it is a case of overflow.
31. 12
32. 9.6
33. The standard library used in formatting numeric output is \( iomanip.h \).
Questions on character and string data

Multiple-choice:
1. (a) ; (b) ; (c) ; (d) ; (e)
2. (a) ; (b) ; (c) ; (d) ; (e)
3. (a) ; (b) ; (c) ; (d) ; (e)
4. (a) ; (b) ; (c) ; (d) ; (e)
5. (a) ; (b) ; (c) ; (d) ; (e)
6. (a) ; (b) ; (c) ; (d) ; (e)
7. (a) ; (b) ; (c) ; (d) ; (e)
8. (a) ; (b) ; (c) ; (d) ; (e)
9. (a) ; (b) ; (c) ; (d) ; (e)
10. (a) ; (b) ; (c) ; (d) ; (e)

Short-answer:
1. Write a declaration for a variable of the type most appropriate to store the value '9'.
2. How are the newline, tab, and null character constants expressed in Java?
3. Name the data type of ‘4’ when it appears in a program.
4. Name the data type of “4.5” when it appears in a program.
5. Declare a character variable and initialize it with the null character.
6. What is the character constant for the newline?
7. What is the name of the table that specifies the numeric encodings of characters?
8. Name four functions defined in the standard library header file string.h.
9. What operation occurs when + or += are used with Java strings?
10. Name a Java standard data type used to store sequences of characters.
11. Name the standard header file that defines the Java class, string.
12. What is a stream?
13. What is a sequence of characters coming from or going to a device?
14. Name a data type for an object used to open and read a file.
15. Show the 8-bit binary value that is stored when a user types R [Enter] in response to the following Java statement, where ch is of type char: cin >> ch;
16. What is the data type of the parameters to toupper and tolower? What is the data type of the values these functions return?
17. How are the newline, tab, and null-character constants expressed in Java?
18. How can you convert an integer to a character value? A character to an integer?
19. What are an advantage and a disadvantage of Java’s automatic type conversion?
20. Write expressions that convert:
   a. “9” to int
   b. 30 to float
   c. “2.8” to float

   d. 0.9 to string
   e. 83 to string
21. Declare and initialize a variable, letter, to store the character value ‘Q’.
22. Declare and initialize a variable, team, to store the string value “Rams”.
23. Declare a string variable, town, with a storage capacity of 40 bytes.
24. What is the data type and memory allocation, in bits and bytes, of a single component of a string data item?
25. How many bytes of memory are allocated by this declaration:
   char prompt[] = “Enter name: “;
26. Declare and initialize a variable, letter, to store the character value ‘Q’.
27. Declare and initialize a variable, team, to store the string value “Rams”.
28. Declare a string variable, town, with a storage capacity of 40 bytes.
29. What is the data type and memory allocation, in bits and bytes, of a single component of a string data item?
30. Declare the integer variables height and weight; initialize them to 67 and 125, respectively; and write a C-style (not cout) output statement to display their values, with appropriate labels.
31. Name the four standard identifiers associated with the Java console input and output statements. For each of the four identifiers, specify its library header file and whether it is a data item (D) or a function (F).
32. Correct the error in this C-language code.
   int input1;
   scanf("%d", input1);
33. Label each term below with the letter of its appropriate definition:
   ______ char
   ______ ASCII
   ______ character string
   ______ fstream.h
   a) an 8-bit data type corresponding to the ASCII table.
   b) A standard table of characters and their decimal encodings.
   c) Storage of characters in consecutive RAM locations.
   d) Standard library file that supplies file input and output object types.
**Long-answer problems**

1. Write a program that displays the second line of a data file, named by the user in response to a prompt. Test it using the name of some program on your work disk; the result should be lines that beginning “//” and describe program examples.

2. One ASCII character produces nonvisual output. Write a program that outputs \texttt{char (7)} and tell what its result is.

3. Write a Java program that uses the \texttt{char} data type name in a type cast to produce the following screen output. Do not use any character or string literals.

   This is a piece of cake

4. Write a program that will print out a five-character ASCII string generated by five integer input values.

5. Write a program that uses a type cast to convert integers to characters and that directly displays an input \texttt{int} variable as an ASCII code.

6. Find out experimentally what the return values of the functions \texttt{isalpha} and \texttt{isdigit} are for character parameters ‘e’, ‘Q’, ‘&’, and ‘4’.

7. Debug this code:

   ```
   ofstream outfile("x.out")
   int x,y;
   cout << "x=" << x << " y= " << y;
   ```

8. Show your initials by assigning ASCII values to character elements of a string variable one at a time and displaying the string.
Study questions on Topic 5: classes

Multiple-choice or T/F:

1. A data item that may have several data attributes and a set of characteristic behaviors is (a) a type; (b) an integer; (c) an object; (d) a class; (e) a control structure
2. (T-F) A structure, unlike an integer, may be compound.
3. (T-F) A member item in a structure type or class is accessible to only one function.
4. Creating new data types is (a) impossible; (b) data abstraction; (c) procedural abstraction; (d) to be discouraged
5. (T-F) A class is an instance of an object.
6. (T-F) Objects are widely considered a factor in making it harder to write longer programs.
7. (T-F) Functions that are members of a class share access to all data members of an object of that class.
8. (T-F) The only ways for different functions to share data in RAM is by use of global variables or parameter passing.
9. (T-F) A class is a data type as opposed to a data item.
10. In a Java program, the identifier part.price could represent (a) a class; (b) an object type; (c) a function header; (d) a member of an object; (e) none of these
11. (T-F) A function may be a member of a class.
12. An instance of this class occupies approximately how many bytes of RAM: class persons {
    char name[30]; int age; 
};
   (a) 0; (b) 2; (c) 30; (d) 34; (e) none of these
13. :: is (a) the scope resolution operator; (b) the delimiter used to access any member of an object; (c) used in all function declarations; (d) a logical operator; (e) not part of Java
14. (T-F) A structure may be passed as a parameter.
15. Up to how many objects may be instances of the same class? (a) 0; (b) 1; (c) 2; (d) several; (e) there is no particular limit
16. (T-F) An identifier declared in braces after enum is a sub-component inside an enumerated-type data item
17. (T-F) With enum we supply a full list of possible values of a data type
18. An enumerated-type value is (a) a compound item; (b) true or false; (c) an int; (d) a float; (e) a char
19. A compound data type may be created with (a) struct; (b) enum; (c) int; (d) for
20. (T-F) A structure value may be passed as a parameter or returned by a function
21. (T-F) A structure may be assigned a value only by assigning values to its members one at a time.
22. A structure within a structure is (a) a syntax error; (b) a logic error; (c) a nested structure; (d) a recursive structure
23. It is often useful to (a) create a structure type especially for a certain function; (b) create a function especially for a certain structure type; (c) declare a structure item without a type name; (d) declare global functions to manipulate global structure variables
24. (T-F) When a class is declared using the class keyword, member items are public, unless specified otherwise.
25. What keyword is used to prevent a member item from being accessed from outside a class's functions? (a) restricted; (b) local; (c) private; (d) public; (e) protected
26. (T-F) A constructor should be called only when the programmer is ready to initialize members of an object with new values.
27. An object may have its members initialized in the statement that declares it, using (a) the initialization operator once for each member; (b) a constructor; (c) an access function; (d) cin; (e) cout
28. (T-F) A program calls a constructor only if the programmer uses its name in a statement.
29. (T-F) An instance of a class may have its members automatically initialized at the time it is declared.
30. (a) ; (b) ; (c) ; (d) ; (e)
31. (a) ; (b) ; (c) ; (d) ; (e)
32. (a) ; (b) ; (c) ; (d) ; (e)
33. (a) ; (b) ; (c) ; (d) ; (e)

Short-answer:

1. If items is a structure type with a float member named price, then write a statement to assign the value 4.99 to an instance of items named notebook.
2. Name a keyword that may be used to declare a data type with which to declare objects.
3. Write a statement that assigns the value 82 to the quantity member of an object, part, which is an instance of class, parts.
4. Write an enumerated type declaration denoting the four seasons and a declaration for a variable, season, of that type.
5. What C++ techniques presented thus far may be used to share data among different functions? Which do the textbook authors prefer?
6. Distinguish between a structure variable and a structure type.
7. Where is a member identifier declared—in a structure variable declaration, in a structure type declaration, or in its own variable declaration?

8. If employee.id_num is an integer, what kind of data item is employee?

9. Is there a syntax error in this program? If so, find it and say how to fix it.

/* persnbug.cpp
Prompts for and displays person name and age.
*/
#include <iostream.h>
struct persons
{
    char name[80];
    int age;
}
void main()
{
    cout << "\nName: ";
    persons person;
    cin >> person.name;
    cout << "Age: ";
    cin >> person.age;
    cout << person.name << " is " << person.age << " years old.\n";
}
10. How are structures and character strings alike, as opposed to the data types int, char, and float?

11. Is the following declaration legal? What if anything is wrong with it?
struct employees
{
    char name[40];
    long salary;
    employees supervisor;
}
12. What is the difference between (a) a structure and an object; (b) a member function and a member variable; (c) a class and an object?

13. How is a time capsule similar to an object that encapsulates data and functions?

14. Write a declaration of a car class, naming some of its features and allowing values to be specified for them.

15. Why is the following code invalid?
struct bits
{
    char name[80];
    bits bit;
};
16. Why is the following code invalid?
struct ID
{
    char name[80];
    int size = 10;
};
17. What is the number of bytes occupied by an instance of the following structure type?
struct employees
{
    char name[20];
    long salary;
};
18. In a program that declares a class children and reads the names and ages of several children from a data file in order to find the average age, will the C++ code that finds the average appear in a class’s member function or in a free function?

19. Why would a programmer wish to declare a new data type?

20. In a program that declared a class employees and an instance emp of that class, would it be appropriate for function employees::display_info to refer to emp? Why or why not?

21. Declare a class circles with members x, y and radius, and write constructors that initialize a variable of type circles to: (a) a screen location of (320,240) and a radius of 100; (b) a screen location and radius specified in the declaration of the variable.

22. Consider the following class declaration:
class employees
{
    public:
        employees();
        employees(char nm[],int hrs);
        char* get_name();
        int get_hours();
        void set_name(char nm[]);
        void set_hours(int hrs);
        void input();
        void display();
    private:
        char name[40];
        int hours;
};
It has:
    a) How many members? _____
    b) How many member data items? _____
    c) How many constructors?

23. Label each term below with the letter of the appropriate definition:
    ______ class
    ______ data abstraction
    ______ encapsulation
    ______ member function
    ______ member variable
    ______ object
    ______ object-based design
    ______ structure
    ______ structure type

(a) A compound data item composed of member items whose types are chosen by the programmer.
(b) A data item that is a component of a structure or object.
(c) A named category of data items that may be used to declare a structure.
(d) The practice of defining new data types.
(e) An object data type, whose instances are defined by their data attributes and their behaviors.
(f) An instance of a class.
(g) The practice of writing plans for programs with an eye to the concepts being modeled, as defined by their data components and behaviors.
(h) The practice of aggregating several data items and separating them from the rest of a program.
(i) A subprogram associated with a class.

**Longer-answer:**

1. Declare a class to represent a machine part, with a name, an inventory quantity, and a price. Declare and define functions to input data and to display the member data values and the part’s current inventory-on-hand value (the product of quantity and price).
2. Write a declaration for a class to represent models of refrigerator. Each model has a price, an identification number, and a name. Write input and display member functions for the class, access functions, and a constructor.
3. Declare a structure type or class for a machine part, with an identification number, an inventory quantity, and a price. Declare, define and call functions to input and display data about a part.
4. Write a program that prompts for prices and sales figures (quantity 0 to 40 thousand) for a product. Define a class of products and store a product as an instance of the class. Write member functions to input and display information about a product, including a bar graph that reflects the quantity sold, in thousands.
5. The data file `children.dat` (below) records names and ages of pupils. Using what you have learned of structures and objects, write a program that reads these four records, displays the names, and displays the average age.

```
Contents of children.dat:
Wendy 13
Jessica 15
Tom 10
Bill 12
```

6. Debug the following program.

```cpp
/* employee.cpp
Prompts for, displays, employee data. */
#include <iostream.h>
struct employees
{
    char name[20];
    long salary;
    int age;
}
```

```cpp
void main()
{
    employees emp;
    cout << "Name,salary," << "age: ";
    cin >> emp.name >> emp.salary >> emp.age;
    cout << "Name: " << emp.name << endl
         "Salary: " << emp.salary << endl
         "Age: " << emp.age << endl;
}
```

7. Declare some C++ classes and define some of their member functions to represent a registration for classes at a college. A registration has a student ID number, a course number, and a date. A date has a month, day, and year.

8. Name some classes and their members, or draw a class diagram, to correspond to the following description: “The Environmental Protection Agency reports to the administration and to Congress. It monitors the activities of industries that may affect the environment and measures levels of certain substances in the air and water.”

9. Write a structure type or class for times of the day that could be expressed on a clock.
10. Write a structure type or class for durations of time using the units of time found in a clock.
11. Write a structure type or class for a date on a calendar.
12. Write a structure type or class for a bank customer’s transaction such as the kind that an ATM could carry out.
13. Write a class, `rationals`, to encapsulate rational numbers. Each rational number has a numerator and a denominator, which are integers. Include member functions `plus`, `minus`, `times`, and `divided by`. Each should accepts a rational-number parameter and return a rational number that expresses the result of the indicated arithmetic operation.
14. Write a set of functions that implement addition, subtraction, multiplication, and division for the structure type `rationals` by accepting a parameter of type `rationals`.
15. Write a function that accepts two instances of type `rationals` and returns a value of type `rationals` that is the product of the two parameters.
Multiple-choice or T/F:
1. c. Types, classes, and control structures are not data items; an integer may not have several attributes.
2. t. A structure may have one or more members.
3. f. A member data item is accessible to all member functions.
4. b. Example: structures and classes (composition).
5. f. An object is an instance of a class.
6. f. Objects are considered a factor in making it easier to write long programs.
7. t. Member functions have automatic access to all data members.
8. f. Two other ways: member functions of a class share data members of that class, and return values pass data back from a function to its calling statement.
9. t. An object is a data item that is an instance of a class. A class is like a template or category.
10. d. The period separates the object name from the member item.
11. t. A member function is declared in the class declaration.
13. a. The scope resolution operator specifies the class of which an identifier is a member, whether a constant, as in ios::fixed, or a member function in the function's definition.
14. t. Use the structure type name to declare the structure parameter.
15. e. A data type may be instantiated any number of times consistent with memory capacity.
16. f. A constant named in an enumerated type declaration is a possible value of an item, not a component.
17. t. As in enum answers {Yes, No, Maybe};
18. c. The values of the constants named in an enum are 0, 1, 2, …
19. a. A structure is a compound item, composed of its members.
20. t. A structure is like a simple-type data item in parameter passing.
21. f. A structure variable may be initialized when declared, with a list of values between braces, or assigned the value of another structure with the assignment operator.
22. c. A structure may have another structure as a member.
23. b. A function that accepts a structure as a parameter in effect implements an operation associated with that structure type.
24. f. Members are private, unless they follow the keyword public.
25. c. A private member is visible only to member functions of its class.
26. f. A constructor is called automatically when an object instance is declared.
27. b. A parameter list may follow the object's name in an object instance declaration. The parameters are passed to the constructor of the object's class.
28. f. A constructor is executed automatically on declaration of an instance of its class.
29. t. A constructor is called automatically and may initialize member items.
30. t. When a structure type is declared with the struct keyword, member access is public by default.

Short-answer:
1. notebook.price = 4.99;
2. struct; class
3. part.quantity = 82;
4. enum seasons {Spring, Summer, Fall, Winter};
5. global variables, parameter passing, member data items
6. A structure variable occupies memory, has a value, and is an instance of a structure type, which does not have a value or occupy memory.
7. structure type declaration
8. employee is a structure or object.
9. error: need semicolon after structure type declaration
10. Structures and character strings are compound, the other types named are simple.
11. Missing semicolon at end
12. (a) A structure, unlike an object, may not call member functions.
### Study questions on Topic 6: Branches and loops

#### Branches

**Multiple-choice or T/F:**

1. > is (a) an assignment operator; (b) an insertion operator; (c) an arithmetic operator; (d) a relational operator; (e) none of these
2. To say "or" in Java, we write; (a) !; (b) &&; (c) ||; (d) ==; (e) "oar"
3. (T-F) The switch statement is a way to avoid multiple nested if's.
4. Which is not a kind of statement? (a) assignment; (b) declaration; (c) input; (d) branch; (e) case label
5. The break statement is used with almost all (a) if statements; (b) loops; (c) switch statements; (d) functions; (e) none of these
6. ! represents (a) disjunction; (b) conjunction; (c) logical negation; (d) abstraction; (e) encapsulation
7. Evaluate a > b && b == c, where a = 2, b = 1, and c = 3? (a) a; (b) 0; (c) 1; (d) 2; (e) none of these
8. Evaluate a != b && a < c, where a = 2, b = 1, and c = 3? (a) a; (b) 0; (c) 1; (d) 2; (e) none of these
9. The case keyword is used in which statement? (a) while; (b) function call; (c) assignment; (d) if; (e) switch
10. && is (a) a logical operator; (b) a relational operator; (c) an arithmetic operator; (d) a Boolean expression; (e) none of these
11. (T-F) (3 > 2 || 1 == 0)
12. (T-F) A block is enclosed by braces.
13. (T-F) A block defines a scope of access
14. (T-F) A block is enclosed in parentheses.
15. (T-F) = is a relational operator.
16. (T-F) > is a logical operator
17. It is normally reasonable to compare items of type _______ for equality. (a) C-style string; (b) int; (c) float; (d) void; (e) none of these
18. Which is not a relational operator? (a) >; (b) <; (c) >=; (d) ==; (e) =
19. Which is not a logical operator? (a) !; (b) ||; (c) <; (d) &&; (e) none is a logical operator
20. A Java type with two possible values is (a) int; (b) double; (c) bool; (d) char; (e) string
21. (T-F) The switch keyword implements the loop control structure.
22. (T-F) The switch statement is used for multiway branches.
23. (T-F) An alternative within a switch statement may have multiple case labels.
24. (a) ; (b) ; (c) ; (d) ; (e)
25. (a) ; (b) ; (c) ; (d) ; (e)
26. (a) ; (b) ; (c) ; (d) ; (e)
27. (a) ; (b) ; (c) ; (d) ; (e)

### Short-answer:

1. Name three C/Java statements that implement the decision control structure.
2. What does || mean in Java?
3. What statement in C or Java takes one of exactly two different courses of action depending on the result of a test of a value?
4. What keyword is used to prevent falling through from one case label to the next in a switch statement?
5. Write an expression in C/Java that tests whether the following is true: either height is greater than 72 or age is not less than 30
6. If you were prompting the user for a number in the range of 1 to 10, and displaying a different message in response to each different number, which C/Java statement would be most recommended?

7. Write a statement that shows an error message if the value of numeric variable `quantity` is negative.

8. Write an expression in C/Java whose value is `true` if `height` is in the range of 60 to 72, otherwise `false`.

9. What is the output of the poorly indented code below? 
   ```
   int x = 1, y = 3;
   if (x < 2)
     if (y > 4)
       cout << "A";
     else
       cout << "B";
   else
     cout << "C";
   ```

10. What keyword is a case label for the condition where no other case label is matched?

11. What are the Java logical operators discussed in this topic?

12. List the Java relational operators.

13. What keyword is used once for almost every case label in a `switch` statement?

14. Write a binary addition table and a binary multiplication table. If `true = 1` and `false = 0`, how do the binary addition and multiplication tables compare with the truth tables for the logical operators `||` and `&&`?

15. Prepare a truth table for a logical binary operator XOR (eXclusive OR) which returns `true` if exactly one of the Boolean operands `A` and `B` is true, and returns `false` otherwise. Do the same for a NAND (Not AND) operator that returns `true` if at least one operand is `false`.

16. Is there a set of braces in every `if` statement? Every `switch`?

17. What is the value of the Boolean expression `A || B || (!A && !B)` where `A` and `B` are Boolean variables? (Try all combinations of values for `A` and `B`.)

18. Write one statement that displays different messages, depending on the value of a character variable named `input`: “Hello” if `input` has the value ‘a’, “One moment” if ‘b’, and “Bye” for ‘c’.

19. Name the syntax error in this statement:
   ```
   if (X > Y)
     cout << X
   else
     cout << Y;
   ```

20. What does the following program output? Indent the nested `if` statements properly.
   ```
   // notgreat.cpp
   #include <iostream.h>
   void main()
   {
     int p = 1, q = 0;
     if (p) if (!q)
       cout << "Y";
     else
       cout << "N";
     else
       if (q)
         cout << "Y";
       else
         cout << "N";
   }
   ```

21. A token is a program element, such as a variable, a keyword, an operator, or a punctuator, that cannot be broken down further into meaningful elements. What Java token always immediately follows the `if` keyword?

22. Distinguish “=” from “==”.

23. Is “a > b > c” a syntax error? What rule does it follow or violate?

24. What Java statement corresponds to the flowchart below?

25. What category of Java statements would correspond to the flowchart below if it were filled in?

26. When would you not use `break` in a `switch` statement?

27. Why would a single set of test data be insufficient to thoroughly test a program containing an `if` statement?
28. Does $2 < 3 \&\& 4 > 1$ mean the same as \(!((2 < 3) \mid\mid (4 > 1))\)? Explain.

29. Assume that $x$, $y$, and $z$ are float variables, is_valid is a Boolean variable, $x = 3.0$, $y = 4.0$, $z = 2.0$, and is_valid = false. Assign a value of true or false to each of the following Boolean expressions.
   a. $(x > z \&\& y > z)$
   b. $(x + y / z) <= 3.5$
   c. $(z > x) \mid\mid (z > y)$
   d. $!$ is_valid
   e. $(x < 1.0) \mid\mid (x >= 3.0)$
   f. $(0.0 < x \&\& x < 3.5)$
   g. $(x <= y \&\& y <= Z)$
   h. $!$ (is_valid \mid\mid $(y + z) >= (x − z))$

30. Debug this error-ridden code:
    ```cpp
    cout << "Do you drink soda pop? ";
    char input;
    cin >> input;
    switch (input)
    {
        'y': cout << "Recycle your bottle\n";
        'n': cout << "Enjoy your juice\n";
        default: cout << "'Y' or 'N' please\n";
    }
    ```

31. What is wrong with this statement?
   ```cpp
   if (ltr == 'a' || 'b' || 'c')
       cout << "Letter is early " << in the alphabet
   ```

32. What is the output of this code? Debug it if you find a logic error.
   ```cpp
   if (x == 2)
       y = 1;
       cout << "same";
   else
       cout << "different";
   ```

33. What is the output of this code? Debug it if you find a logic error.
   ```cpp
   int n = 3;
   if (n == 2)
       cout << "2"
   else
       cout << "not 2"
   ```

34. What is the output of this code? Debug it if you find a logic error.
   ```cpp
   char name[40];
   cout << "Name ";
   cin >> name;
   if (name == "")
       cout << "Is blank"
   ```

35. What do indents mean in Java syntax?

36. Label each term below with the letter of its appropriate definition:
    - Boolean expression
    - branch
    - break
    - case label
    - conjunction
    - conditional statement

37. List and identify (a) the C/Java relational operators; (b) the logical operators.

38. Write one C or Java expression that is true in cases where both of two conditions hold: (i) $a$ is at least 3; (ii) the value of answer is 'Y'. Assume that $a$ has been declared as an int and answer as a char.

39. Write a statement that tests whether a variable, quantity, is over 10, and if so reduces the value of another variable, price, by 5% and displays the message “discounted”
40. Evaluate these C/Java expressions:
   ___ ! (2 > 1)
   ___ 3 < 2 || 9 > 5
   ___ 1 + 1 == 2 && 4 != 5

 Longer answer problems:
1. Write a program that accepts a distance in miles and displays “OK” if it is not higher than 100, otherwise “Too far.”
2. Write a program that accepts three floating-point numbers and displays the smallest and largest. For full credit, show output verifying that your logic is correct for all cases.
3. Write a program to input two real numbers and to compute and display the absolute value of the difference between them. Use an if statement rather than a function to do this.
4. If a program performs a division operation with zero as the divisor, it will terminate with an error message. Write a program that prompts for two integers and displays their quotient, showing its own error message if the divisor is 0.
5. Write a program that sets a Boolean variable is_factor to true if the second of two input integers is a factor of the first. Otherwise is_factor is set to false.
6. Write a program that allows the user to input any two real numbers and then choose, using a switch statement, one of the four operations +, −, *, or /. The computed result, correct to a number of decimal places specified by the user, will be displayed. If the second input number is 0.0 and the operation “/” is chosen, the message “Can’t divide by zero” should be displayed instead of a computed result.
7. Write a program to convert a letter of the alphabet to the corresponding telephone dial digit, based on the following conversion table:

   Letter   Dial digit
   'A'..'C'  2
   'D'..'F'  3
   'G'..'I'   4
   'J'..'L'   5
   'M'..'O'  6
   'P','R','S' 7  {Q is not included on telephone dial}
   'T'..'V'   8
   'W'..'Y'   9  {Z is not included on telephone dial}

8. If a non-convertible ASCII character is input, print an appropriate error message.
9. Write a program to classify a person on the basis of input height and weight. Use the following classification scheme:

   Height  Weight  Classification
   > 72 in. > 190 lb.   Tall and heavy
   > 72 in. ≤ 190 lb.   Tall and light
   ≤ 72 in. > 170 lb.   Short, heavy
   ≤ 72 in. ≤ 170 lb.   Short, light

10. Write a program that prompts for an integer from 0 to 99 and accepts it into a string variable. Convert it to an integer variable without using atoi and display the integer variable’s value. Display an error message if the input string is longer than two characters or if any character input is not a digit.
11. Debug the following program.

   // weighbug.cpp
   // Asks weight, categorizes
   // user according to weight.
   // HAS BUGS.
   #include <iostream.h>

   void main()
   {
      cout << "Your weight? ";
      int weight;
      cin >> weight;
      if (weight == 0)
         cout << "Invalid ";
      else
         cout << "You are ";
         switch(weight / 100)
         case 0:
            cout << "light
";
         case 1:
            cout << "normal
";
         case 2,3,4:
            cout << "heavy
";
   }

12. Write a program to prompt for a letter of the alphabet and use a switch statement to convert it to the corresponding telephone dial digit, based on the following conversion table:

   A B C 2
   D E F 3
   G H I 4
   J K L 5
   M N O 6
   P R S 7
   T U V 8
   W X Y 9

   Be sure to validate input.

13. Write and test a program that prompts for six floating-point numbers and displays the largest. You will need to use only one relational operator, five times.
14. Simplify this code:
char input;
cin >> input;
switch(input) {
    case '0':
        cout << "digit"; break;
    case '1':
        cout << "digit"; break;
    case '2':
        cout << "digit"; break;
    case '3':
        cout << "digit"; break;
    case '4':
        cout << "digit"; break;
    case '5':
        cout << "digit"; break;
    case '6':
        cout << "digit"; break;
    case '7':
        cout << "digit"; break;
    case '8':
        cout << "digit"; break;
    case '9':
        cout << "digit"; break;
}

Answers to study questions on branches

Multiple-choice or T/F:

1. d. The other relational operators are <, ==, <=, >=, !=.
2. c. The OR logical operator takes two logical operands.
3. t. A cascade of if s is replaced in switch by a list of case labels and associated statements.
4. e. A case label is part of a switch (branch) statement.
5. c. In a switch statement, a break is needed after the statements under a case label to prevent the statements following later case labels from executing.
6. c. The Java not operator is "!" , which turns all zero values to 1 and all nonzero values to zero.
7. b. The expression is false because b (1) is not equal to c (3). False evaluates to 0.
8. c.
9. e. A switch statement should contain case labels, followed by statements that will execute when a match to the switch selector is found.
10. a. The && operator is the logical operator of conjunction, AND.
11. t. 3 is greater than 2, and the OR operator returns true if either operand is true.
12. t.
13. c. Statements and expressions are not tokens; the left parenthesis is mandatory.
14. c. The two equal signs are a single relational operator.
15. d.
16. f. The keyword else is part of the if statement and should be preceded by a semicolon.
17. b. The if statement is a selection statement.
18. b. The test fails on both terms.
19. a. || is the logical operator of disjunction; OR
20. b. != is the relational operator for inequality.

Short-answer:

1. if, if...else, switch
2. or
3. if ... else
4. break
5. height > 72 || age >= 30
6. switch
7. if (quantity < 0)
   cout << "Invalid value\n";
8. height >= 60 && height <= 72
9. BC
10. default
11. !, ||, &&
12. <>, <=, >=, !=, ==
13. break
14. 
15. 
16. 
17. 
18. switch (input) 
   {
   case 'a':
        cout << "Hello";
        break;
   case 'b':
        cout << "One moment";
        break;
   case 'c':
        cout << "Bye";
   }
19. 
20. 
21. 
22. 
23. 
24. 
25. 
26. 
27. 
28. 
29. 
30. 
31. 
32. 
33. 
34. 
35. 
36. 
37. (a) ==, !=, <, >, <=, >= (b) !, ||, &&
38. a >= 3 && answer == 'Y'
39. if (quantity > 10) 
   {
   price *= 0.95;
   cout << "discounted";
   }
40. 0 (false); 1 (true); 0 (false)
Loop statements

Multiple-choice or T/F:
1. The most appropriate statement for reading data from a file is (a) \textit{if}; (b) \textit{while}; (c) \textit{do}; (d) \textit{switch}; (e) assignment
2. The \textit{while} statement is (a) counter driven; (b) bottom tested; (c) recursive; (d) top tested; (e) to be avoided
3. Counters are used (a) in all loop statements; (b) in all well-written loop statements; (c) normally in \textit{for} statements; (d) only in \textit{for} statements
4. \textit{++} is (a) a binary operator; (b) the increment operator; (c) the decrement operator; (d) an operator that has no effect on its operand; (e) a syntax error
5. An infinite loop (a) is the goal of every programmer; (b) is generally a logic error; (c) is very rare; (d) can be fixed by inserting a semicolon before the loop body; (e) none of these
6. (T-F) In a bottom-tested loop the body is always executed at least once.
7. What appears after the \textit{do} keyword is (a) a Boolean expression; (b) always a compound statement; (c) a statement; (d) a parenthesis
8. The \textit{for} statement implements (a) a counted loop; (b) a top-tested loop; (c) a multi-way branch; (d) a recursive function
9. In a \textit{for} statement the parentheses after \textit{for} contain in order (a) an exit condition, then an update, then an initialization; (b) an update, an exit condition, and an initialization; (c) an initialization, an exit condition, and an update; (d) an initialization, an update, and an exit condition; (e) any of the above
10. (T-F) It is possible to use any of the three Java loop statements to solve a given problem that involves repetition.
11. Which is not favored in the world of structured programming? (a) the middle-tested loop; (b) the top-tested loop; (c) the bottom-tested loop; (d) simple branches; (e) nested branches
12. The \textit{break} statement terminates (a) a program; (b) a function; (c) the current loop or branch; (d) all loops or branches; (e) none of these
13. (a) ; (b) ; (c) ; (d) ; (e)
14. (a) ; (b) ; (c) ; (d) ; (e)
15. (a) ; (b) ; (c) ; (d) ; (e)
16. (a) ; (b) ; (c) ; (d) ; (e)

Short-answer:
1. Write a C or Java statement that displays the numbers from 100 to 200.
2. Name three Java loop statements.
3. How many asterisks would the code below display?
   \begin{verbatim}
   for (int i=0; i < 40; ++i)
   for (int j=0; j < 10; ++j)
   cout << "+";
   \end{verbatim}
4. What is the output of this code?
   \begin{verbatim}
   for(int i=10; i < 16; ++i)
   if (i % 5 != 0)
   printf("*");
   \end{verbatim}
5. How many asterisks are output by this code?
   \begin{verbatim}
   for(int i=0; i < 40; ++i)
   cout << "*";
   for(int j=0; j < 10; ++j)
   cout << "*";
   \end{verbatim}
6. Which C/Java keyword or keywords would be most convenient to implement the design pictured in the flowchart below?

7. What are the three items in parentheses just after the keyword \textit{for} in a C/Java program?
8. What must a \textit{while} loop contain if its body has more than one statement?
9. What is the result if a loop exit condition cannot be fulfilled?
10. Name the three Java loop statements, using keywords and brief (two-word) descriptions.
11. How many statements comprise the loop body in each Java loop statement?
12. What is the Java code in parentheses after \textit{while} for?
13. What does the Java code after the second semicolon in \textit{for} statement do?
14. Which Java statement would be most appropriate for getting user input until the user enters a blank string?
15. Rewrite the following statement as a \textit{while} loop; as a \textit{do...while} loop:
   \begin{verbatim}
   for (int i=1; i < n; ++i)
   cout << setw (i * i) << '*';
   \end{verbatim}
16. What would you do to ensure that a program that uses a loop to read file data will not fail due to the inability to open the file?
17. A simple \textit{for} loop has a nesting level of 1; a \textit{for} loop within a \textit{for} loop has a nesting level of 2; a \textit{for} loop within a \textit{for} loop within a \textit{for} loop has a nesting level of 3; and so on.
18. How many iterations do nested for loops execute if the nesting level is 3 and each nested loop’s control variable has an initial value of 0 and a final value of 9?
19. What if the nesting level is m and each nested loop’s control variable has an initial value of 0 and a final value of n−1?
20. What predefined Java function would you use to simulate a process that produces a sequence of unpredictable numbers, such as a list of winning lottery numbers?
21. The code below is intended to display the numbers from 1 to 10. (a) Will it compile? (b) Does it accomplish its objectives? If not, correct it.
   ```java
   int a = 1;  
   while (a < 10) {  
     a = a + 1;  
     cout << a << endl;  
   }
   ```
22. What is the output of the following program code?
   ```java
   int x = 10;  
   while (x > 0) {  
     x = x - 3;  
     cout << x << endl;  
   }
   ```
23. Here is a program fragment that was written to find the sum of all positive integers less than n, where n is input from the keyboard. Does it work? Explain your answer.
   ```java
   int n;  
   cout << "Number? ";  
   cin << n;  
   if (n > 0) {  
     int sum = 0;  
     int i = n - 1;  
     while (i > 0) {  
       sum = sum + i;  
       cout << "The sum up to " << n << " is " << sum;  
     }  
   }
   ```
24. What is the risk in using a do...while loop to read data items from a text file?
25. What is wrong with the following main function, intended to compute 2^n?
   ```java
   void main() {  
     int power;  
     while (power < 1000) {  
       cout << power << endl;  
       power = power * 2;  
     }
   }
   ```
26. (Challenge Exercise) Construct a flowchart of the logic of the for loop.
27. Debug this code, which compiles to an error message, “Multiple declarations of m”:
   ```java
   for (int m = 0, m < 10, ++m)  
     cout << m;
   ```
28. Debug this code:
   ```java
   int num = 5, i;  
   for (i=0; i < num, ++num)  
     cout << i;
   ```
29. How many stars does the code below output? How many if the braces are removed?
   ```java
   for (int i=0, i < 5, ++i)  
     {  
       cout << " ";  
       for (int j=0; j < 10; ++j)  
         cout << " ";  
     }
   ```
30. Label each term below with the letter of its appropriate definition:
   - bottom tested
   - break
   - control variable
   - counted loop
   - do...while
   - end-of-file
   - file pointer
   - for
   - ifstream::eof
   - ifstream::get
   - infinite loop
   - iteration
   - nested loop
   - ofstream::put
   - return
   - top tested
   - while
   a) repetition
   b) loop that iterates a predetermined number of times
   c) top-tested C++ loop statement
   d) loop without working exit condition
   e) single-character file input routine
   f) single-character file output routine
   g) state of affairs when all available characters have been read
   h) function used to tell when to exit file-reading loop
   i) system value used to keep track of where next access should occur
   j) loop whose exit condition is before body
   k) loop whose body is before exit condition
   l) bottom-tested C++ loop statement
   m) data item used to count a loop’s iterations or otherwise set exit condition
   n) unstructured statement enabling exit from loop but not function
   o) unstructured statement enabling exit from function
   p) loop within a loop
   q) C++ keyword for counted loop
31. Name the three Java loop statements, using keywords and brief (two-word) descriptions.
32. What does the Java code after the second semicolon in the for statement usually do?
33. Label each term below with the letter of its appropriate definition:
   _____ logic error
   _____ runtime error
   _____ specification error
   (a) Mistaken instructions to programmer concerning what a program should do
   (b) Bug
   (c) System message reporting division by zero, invalid operand to a function, etc.
34. Does the code below, when included in a program, generate a compiler error? If not, what does happen? Can you explain the observed results?
   ```
   int a = 1;
   while (a < 10);
   a = a + 1;
   cout << a << endl;
   ```

1. Name the three Java loop statements, using keywords and brief (two-word) descriptions.
2. What does the Java code after the second semicolon in parentheses in the for statement usually do?
1. Label each term below with the letter of its appropriate definition:
   _____ logic error
   _____ runtime error
   _____ specification error
   (a) Mistaken instructions to programmer concerning what a program should do
   (b) Bug
   (c) System message reporting division by zero, invalid operand to a function, etc

**Longer answer problems:**
1. Write a program that repeatedly prompts for integer values and displays the largest one. Terminate the input on entry of a zero value.
2. Write a program that prompts for a loan amount for a prime-rate loan at 8.5%, and displays the amount to be repaid after 30 years, including principal and interest. Show results accurate to the cent. For numeric values use named constants. On input of a zero or negative amount, display an error message.
3. Write a program that repeatedly accepts a numerical score from 0 to 100 and displays a letter grade, according to the rule: 0-59 is ‘F’, 60-69 is ‘D’, 70-79 is ‘C’, 80-89 is ‘B’, and 90-100 is ‘A’. Terminate the loop on input of a negative score and display an error message on invalid input.
4. Write a program that prompts for 100 numbers and displays the largest one.
5. Write a program that prompts for integers, until the user enters 0, and displays each one, except the zero.
6. Write a program that repeatedly prompts for integer values and displays the largest one. Terminate the input on entry of a zero value.
7. Write a program that displays all the numbers from 1 to 1000 that are divisible by both 5 and 3.
8. Write a program that repeatedly prompts for three ages and displays their average. Display an error message if invalid input is entered. Repeat the prompt-input-display process until the first input value is zero.
9. Write a program that prompts for two characters and displays all characters in the range from the first to the second. For example, input of ‘A’ and ‘E’ should yield output of “ABCDE”.
10. Write a program that displays this figure once, using a loop:
   ```
   X
   XX
   XXX
   XXXX
   ```
11. Write a program that repeatedly prompts for a character and displays the string “yes” if the input is the letter Y, “no” if N, and “maybe” if M, and an error message otherwise.
12. Write a program that repeatedly prompts for integer values and displays the largest one. Terminate the input on entry of a zero value.
13. Implement this flowchart as a C or Java program:
   ![Flowchart Image]
14. Implement this flowchart as a C or Java program:

```
Prompt for price, depreciation rate, expected life
value ← price
years remaining ← expected life

years remaining ≥ 0 ?
T
F

depreciation ← value × depreciation rate
Reduce value by depreciation rate
Reduce years remaining by 1

Display value
```

15. Write a program that displays a logarithm table, showing the natural logarithms of integer values from 1 to 100.

16. Write a program that displays the screen graphics characters and their ASCII codes. The graphics characters are those with values 128 to 255 in the ASCII table.

17. Input pairs of real numbers (e.g., \( r_1 \) and \( r_2 \)). Display the absolute value of the difference, \( |r_1 - r_2| \), for each input pair. Exit from the loop if \( r_1 < 0 \).

Sample output:

```
r_1  r_2  |r_1 - r_2|
5.00  16.90  11.90
21.50  13.25   8.25
```

18. Input real numbers \( a \) and \( b \). Compute and display \( (a-b)/(a+b) \). Use a while loop to test input values to be sure \( (a+b) \neq 0 \) before attempting computation. If \( (a+b) = 0 \), input new values for \( a \) and \( b \).

Sample displayed input:

Enter two real values whose sum is not 0: 23.5 - 23.5
Enter two real values whose sum is not 0: 23.5 10.6543

Corresponding output:

23.500
10.654
\( (a-b)/(a+b) = 0.376 \)

19. Loop to input Celsius (Centigrade) temperatures and print corresponding Fahrenheit temperatures until 999.0 is input. The formula for converting Celsius values to Fahrenheit values is, Fahrenheit = 1.8 * Celsius + 32

20. Write a program that has no input and outputs the following:

```
abcde
bcdef
cdefg
defgh
efghi
```

using nested loops.

21. Write a program that inputs values until the user enters 0 and outputs “Ascending” if each value before the last is greater than or equal to its predecessor; otherwise it outputs, "Not ascending."

22. If you invest $1.00 today and the investment accumulates 5% interest each year for 100 years, what will it yield to your great-great-grandchildren then? Write a program to compute the result.
Answers to study questions on loops

Multiple-choice or T/F:
1. b. The *while* loop can test for end-of-file at the start, as is necessary.
2. d. The test occurs before the body of the *while* loop.
3. c. Any loop statement may have a counter; the *for* statement is designed especially to support counted loops.
4. b. The *++* operator adds 1 to the value of its variable operand.
5. b. Any loop should have an exit condition that is able to become *true*.
6. t. The exit condition is tested after the body, so the body will execute at least once.
7. c. The body of a *do*…*while* loop may be a simple or compound statement.
8. a. The counter variable is manipulated by the expression-statements in parentheses after *for*.
9. c. The three expression-statements are separated by semicolon and their order is significant.
10. t. Any of the three loops may be used to solve a given problem; which one is best is a matter of convenience.
11. a. A loop should test and exit either before or after the execution of the body.
12. c. To exit a loop or a *switch* statement, we use *break*.

Short-answer:
1. `for (int i = 100; i<= 200; ++i)
   cout << i << " ";`
2. *while, do*…*while, for*
3. 400 asterisks
4. ****
5. 50 asterisks
6. *do*…*while*
7. A *for* loop has in parentheses an initialization, an exit/continuation test, and an update.
8. The loop must have a compound statement as its body.
9. The loop will be infinite.
Study questions on Subprograms

Multiple-choice or T/F:

1. (T-F) A function body must contain the name of the function.
2. (T-F) The void data type is used when it is not known what the user will input.
3. (T-F) A function declaration specifies what the function does in detail.
4. (T-F) Procedural abstraction consists of breaking down a problem or program into simpler parts.
5. How many definitions must a function have in a program? (a) 0; (b) exactly 1; (c) 2; (d) at least one
6. A way to make a program more modular is to (a) document variables with comments; (b) print clear output; (c) use file input; (d) write function definitions
7. The top-down approach (a) breaks down a problem and solves it step by step; (b) begins with ready-made components and puts them together; (c) focuses on minimizing the number of functions; (d) originated with object-oriented programming
8. (T-F) According to the text, a single function should perform a wide variety of tasks.
9. A function prototype (a) introduces the function name to the program; (b) is a declaration; (c) must occur before the function is used; (d) all of the above; (e) none of the above
10. Reasons to define functions include (a) maximizing the number of identifiers and reducing code repetition; (b) modularity and maximizing the number of identifiers; (c) modularity and reducing code repetition; (d) reducing code repetition and increasing the size of programs; (e) modularity and increasing program size
11. We are urged to (a) avoid local variables; (b) avoid global variables; (c) maximize side effects; (d) avoid global constants; (e) none of these
12. Local variables are located (a) within the function's machine code; (b) in the same area as global variables; (c) on the stack; (d) in the microprocessor
13. A function definition (a) is a function declaration; (b) has a header and a body; (c) terminates with a semicolon; (d) may be used as a program statement; (e) none of these
14. A value may be passed out of a function to the calling statement with a (a) value parameter; (b) reference parameter; (c) variable parameter; (d) goto statement
15. (T-F) All parameters and variables declared by name are passed using the stack.
16. (T-F) A formal parameter appears in parentheses in a function call.
17. With a value parameter, what is passed to the function? (a) the address; (b) the value of an expression; (c) the full text of the expression; (d) nothing; (e) a request for information
18. A function whose name is used as an expression in a program should have (a) a value parameter; (b) a reference parameter; (c) a return value; (d) none of these; (e) all of these
19. Which would you use to cause a function to prompt for two quantities and pass them back to the calling statement? (a) a loop; (b) value parameters; (c) reference parameters; (d) return values; (e) a class
20. In a function call we might find (a) a function definition; (b) a function header; (c) a formal parameter; (d) an actual parameter; (e) none of these
21. Which operator is used to declare a parameter by reference? (a) none; (b) ref; (c) &; (d) *; (e) !
22. According to the text, the best way to share temporary data items between functions is (a) class members; (b) global variables; (c) local variables; (d) parameters; (e) all of the above
23. Parameter values are stored (a) with the calling function's machine code; (b) with the called function's machine code; (c) in the calling function's activation record on the stack; (d) in the called function's activation record on the stack; (e) in a program's variable memory
24. The stack is like a (a) bulletin board; (b) rope; (c) bookshelf; (d) chain; (e) none of these
25. (T-F) A given identifier may be used in only one function declaration.
26. (T-F) The values of all parameter expressions are copied from the calling function into the called function.
27. A function that returns a value (a) must do so with a parameter; (b) should specify the data type of that value in the function header. (c) does so with an assignment statement; (d) does so automatically
28. (T-F) A recursive function should have an if or switch statement in it.
29. (T-F) A recursive function is one that calls itself.
30. Which is not a kind of loop statement in Java? (a) repeat; (b) counter-driven; (c) top-tested; (d) bottom-tested; (e) recursive
31. Recursion is (a) a way to write a loop; (b) a kind of file input; (c) a nested loop statement; (d) used in all loops
32. (T-F) A function call should contain a data type name.
33. (T-F) A function prototype should contain a data type name.
34. (T-F) A function definition should contain a data type name.
35. (a) ; (b) ; (c) ; (d) ; (e)
36. (a) ; (b) ; (c) ; (d) ; (e)
37. (a) ; (b) ; (c) ; (d) ; (e)
38. (a) ; (b) ; (c) ; (d) ; (e)

Short-answer:
1. What are two ways that a function can pass a data value back to the statement that called it?
2. Write the prototype for a function that accepts two float parameters and returns their integer sum.
3. Write an appropriate declaration for a function that accepts two integers, returns no value, and draws a figure. You need not spell out everything that the function does.
4. In the module hierarchy diagram below, (a) name functions that call show_results; (b) name functions that are called by calculate; (c) does get_input call show_results? (d) does find_costs call calculate? (e) does main call find_profit? (f) does calculate call find_profit?

5. What tools does an application programmer have in addition to the keywords and operators of the C++ language?
6. Make up names for three functions that might be used in a program that consists of these three modules: (a) Input an item purchase price from the keyboard; (b) Compute a sales tax and a total sale amount for the purchased item; (c) Display purchase price, tax amount, and total sale amount.
7. What is missing from this function definition?
   ```cpp
   void fetch_price()
   cout << "Enter price: "
   cin >> price;
   ```
8. Give two reasons for using a modular approach to program development.
9. What is the difference between a function declaration and a function definition?
10. What must be true of calls to functions with the void type designation?
11. Name a technique used in debugging to pinpoint the source of an incorrect output value.
12. What is one way to share data among functions without using parameters or return values? What are its disadvantages?
13. Put these phrases or subphrases of the problem-solving process in chronological order, numbering the first “1”, the second “2”, etc.
14. _____ code program
15. _____ desk check
16. _____ write a design
17. _____ get problem specifications
18. _____ test program
19. _____ debug code
20. Draw a module-hierarchy chart for the following skeleton program.
// prog.cpp: Does nothing
void fa();
void fb();
void fc();
void fd();
void main()
{ fa(); fd(); }
void fa() { fb(); fc(); }
void fb() {}
void fc() {}
void fd() {}

21. Do the C++ statements in a function always execute at least once when the program containing the function executes? Justify your answer.

22. Find the syntax or other errors in the programs below
(a) // hello2.cpp
#include <iostream.h>
void say_hello();
void main() { say_hello(); }
void say_hello() {}
(b) // hello3.cpp
#include <iostream.h>
void say_hello();
void main() { say_hello(); }
void say_hello() {}
(c) // hello4.cpp
#include <iostream.h>
void say_hello();
void main() { say_hello(); }
void say_hello() {}
(d) // hello5.cpp
#include <iostream.h>
void say_hello();
void main() { say_hello(); }
void say_hello() {}
(e) // hello5.cpp
#include <iostream.h>
void say_hello();
void main()
{ // says hello
  say_hello();
}
void say_hello()
{ // This function is empty

23. Could a void function contain a return statement, and if so, what would it look like?

24. In program example x.cpp (Problem 18 above), name:
(a) An actual value parameter;
(b) An actual variable parameter;
(c) A formal value parameter;
(d) A formal variable parameter.

25. What is the difference between an actual parameter and a formal parameter?

26. What is the difference between a reference parameter and a value parameter?

27. What information other than a parameter name is required in a parameter declaration?

28. Could the expression (age - 2) be a reference parameter? Explain.

29. What can you conclude about two variables or parameters, used in two different functions, if the data items have the same identifier?

30. When a reference parameter of type char is passed, how many bytes of data are copied to the stack and communicated to the called function?

31. Name one positive aspect of recursion as opposed to iteration, from a programming point of view, and one negative aspect from a machine-operation point of view.

32. What operator, if any, is used to declare (a) a reference parameter; (b) a value parameter.

33. What will the compiler produce from the following code?
// thunk.cpp
#include <iostream.h>
const int ZIP = 01776;
void blip(int& jabr);
void main()
{ blip(ZIP); }
void blip(int& jabr)
{ JABR = -JABR; }

34. What is wrong here?
// sub.cpp
```cpp
#include <iostream.h>
void subtract(int a, int b, int& diff);
void main()
{
    subtract(2, 3, -1);
}
void subtract(int a, int b, int& diff)
{
    diff = a - b;
}
```

35. What would be a good declaration for a function that takes parameters and that displays the interest amount on a loan, given principal, interest, and term of loan?

36. What is wrong with this function declaration:
```cpp
void draw_rectangle(int width, ht);
```

37. What is wrong with this recursive function?
```cpp
void get_amt()
{
    cout << "Sales amt: ";
    float amt;
    cin >> amt;
    get_amt();
}
```

38. What is a difference between the body of any void function and the body of a non-void function?

39. Label each term below with the letter of its appropriate definition:

   - debugging
   - trace statement
   - function
   - function declaration
   - function definition
   - global variable
   - intermediate variable
   - local variable
   - module
   - procedural abstraction
   - recursion
   - activation record
   - actual parameter
   - base case
   - formal parameter
   - function call
   - function declaration
   - parameter passing
   - pass-by-reference
   - pass-by-value
   - recursive case
   - return value
   - stack

   a) Code that spells out all steps taken by a C++ subprogram.
   b) Giving a name to a series of program statements, for use in a program.
   c) Finding and fixing logic errors.
   d) A data item not input or output.
   e) Code that allows a programmer to see values of variables while debugging.
   f) A data item accessible to all functions in a program.
   g) A data item accessible only to function that declares it.
   h) A prototype, introducing identifier to a program.
   i) A C++ subprogram.
   j) Use of a parameter that may communicate a new value back to the calling statement.
   k) Mechanism for passing a value to calling statement using the function call as an expression.
   l) Introduction of subprogram name to a program.
   m) Argument as found in function header.
   n) Data item that stores a function invocation in memory.
   o) Nonrecursive branch in recursive function.
   p) Language mechanism for communicating data between functions.
   q) Use of a parameter that communicates data into called function but not out.
   r) Invocation of subprogram.
   s) Argument found in function call.
   t) Attribute of a function that calls itself.
   u) Memory structure for sharing data between functions.
   v) Mechanism for automatically initializing member data items.
   w) Statement in recursive function in which function calls itself.
Longer answer problems:
1. Write a program that defines and calls two functions: one to prompt for, input, and return the size of a square, and the other to draw a rectangle, composed of X’s like that below. Your drawing function should accept parameters for both the height and the width of the figure. For a square, the width and height are the same.
   Sample I/O:
   Size of square: 3
   XXX
   XXX
   XXX

2. Write the definition of a function that accepts the positions of two opposite corners of a rectangle as parameters and returns the area of the rectangle.

3. Write a program that reads a text file composed of lines and displays it on the screen.

4. Write a function that accepts an integer parameter, length, and displays a line of that many hyphens.

5. Write the definition of a function that accepts two integer values as arguments and that returns the smaller one.

6. Break down the following program into three functions, input, calculate, and display, each called from main. Use parameters rather than global variables to share data among functions.

   // final.cpp
   #include <iostream.h>
   void main()
   { int in1,in2;
   do {
   cout << "Enter 2 " << "non-negative integers:";
   cin >> in1 >> in2;
   } while(in1 < 0 || in2 < 0);
   int product = 1;
   for (int i=0; i < in1; ++i)
   product *= in2;
   cout << in1 << " to the " << in2 << " power is " << product << endl;
   }

7. Write a module hierarchy chart for a program that inputs two numbers from the user, a and b, calculates the value $a^b$ (a raised to the b power), and displays the results of the calculation. You do not need to write a program, but your answer should include meaningful module names and show how the modules are related to each other. Declare function prototypes, including parameter declarations.

8. Write a program that defines and calls two functions: one to prompt for, input, and return the size of a square, and the other to draw a rectangle, composed of X’s like that below. Your drawing function should accept parameters for both the height and the width of the figure. For a square, the width and height are the same.
   Sample I/O:
   Size of square: 3
   XXX
   XXX
   XXX

9. Write a program that declares, calls, and defines a function that accepts three integer values as parameters and returns the smallest one.

10. Use two functions, one to draw a horizontal line of eight asterisks and the other to draw two asterisks separated by six spaces on the same line.

11. Modify your solution to Exercise 1 to include a new function that calls the other two. Each time the new function is called, a box is drawn.

12. Write a function that repeatedly prompts for and inputs integer values, until the user enters a zero value, and displays the smallest one and the average.

13. Write a program containing a function, get_food, that asks the user for a name of a food item to be input from the keyboard, and writes the name to a disk file. Call the function four times to put a meal together. Print the meal file.

14. Here is a structure diagram of a modularized program that adds four input values and shows the sum:

   ![Structure Diagram]

   Implement this design, using a function for each of the three indicated modules. Note that the function for inputting a single value and adding it to the sum will need to be called more than once. Document your program with comments, user prompts, and clearly identified output.

15. State income tax for the State of Panic is computed according to the following formula:
3% on net annual income up through $8000.00, plus
5% on net income in the range $8000.01 – 15000.00, plus
8% on net income in excess of $15000.00.
Net income is gross income minus deductions.

16. Use a while loop to permit input of multiple sets of data (name, gross sales, and deductions). Use separate functions for (1) data input, (2) tax computation, and (3) a tax report written to the screen.
17. Use separate functions to compute the area of a square and the area of a circle using the same input real value as side and diameter, respectively. Use the \( M \_PI \) constant (See Chapter 3) in your calculation of the area of the circle. Use a \( do...while \) loop to permit multiple entries. The area of a circle of radius \( r \) is equal to \( \pi r^2 \).

18. The faculty of Commuter Community College have been voted an across-the-board 5.5% pay increase. Input an ID number and old salary for each professor. Use a single function to: (a) compute a new salary; (b) keep a running total of the old salaries; and (c) keep a running total of the new salaries. Use no global variables. Display a salary report similar to the following sample:

<table>
<thead>
<tr>
<th>ID</th>
<th>OLD SALARY</th>
<th>PAY RAISE</th>
<th>NEW SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joe</td>
<td>23000.00</td>
<td>7000.00</td>
<td>30000.00</td>
</tr>
<tr>
<td>Tucker</td>
<td>16000.00</td>
<td>670.00</td>
<td>16670.00</td>
</tr>
<tr>
<td>Linda</td>
<td>25000.00</td>
<td>15000.00</td>
<td>40000.00</td>
</tr>
<tr>
<td>Carey</td>
<td>10000.00</td>
<td>340.00</td>
<td>10340.00</td>
</tr>
</tbody>
</table>

8000.00 240.00

19. Write a program to input an integer \( a \) and an integer \( b \). Use a function to compute \( ab \). Include a loop in the function \( main \) to allow for multiple sets of input data. Use an input of \( a = 0.0 \) as a signal value to exit the loop. Remember that \( a0 = 1 \) for \( a = 0 \).

20. Use a function to compute the sum of the first \( n \) terms of the geometric series:

\[
x + ar + ar^2 + ar^3 + \ldots + ar^{n-1}
\]

where \( a \) (real), \( r \) (real), and \( n \) (integer) are input from the keyboard. The function should accept three parameters.

For example, if \( a = 2 \), \( r = 3 \), and \( n = 4 \), then the function should return \((2 + 2*3 + 2*3*3 + 2*3*3*3) = 80.0\).

21. The number sequence \(1, 1, 2, 3, 5, 8, 13, \ldots \) is known as the Fibonacci sequence. The first and second terms are both 1; the third and all subsequent terms are the sum of the two immediately preceding terms. For example, the 7th term is the sum of the 5th and 6th terms.

22. Write a program to input an integer \( n \), where \( n \geq 1 \); compute and display the first \( n \) terms of the Fibonacci sequence. Use a function to compute and display the third and all subsequent terms.

Sample I/O:
Enter number of Fibonacci terms desired (n > 0): 8
1 1 2 3 5 8 13 21

23. Input an unsorted list of integers. Let the user enter a sentinel value of 999 to terminate input. Use a single function to pass back to \( main \) both the largest and smallest values in the list. Don’t include the sentinel value (999) in the output, or for consideration as largest or smallest.

24. (Challenge) Input two positive integers. Use a function to compute the greatest common divisor (GCD). The greatest common divisor of two integers is the largest integer that is a factor of both integers. Include a loop for multiple sets of input. Use a sentinel value of 0 to exit the input loop.

Sample I/O:
Enter first integer (0 to exit): 19626
Enter second integer: 20184
Greatest common divisor: 6

Enter first integer (0 to exit): 2001
Enter second integer: 2001
Greatest common divisor: 15

Enter first integer (0 to exit): 0

25. Descriptive statistics are grouped into two categories, measures of central tendency and measures of dispersion. Measures of central tendency tend to locate the middle of a set of data. Common measures of central tendency are:

- The mean: The sum of a set of \( n \) values, divided by \( n \). (The mean is commonly referred to as the 'average'.)
- The median: The middle value when the data is placed in order of size. The position of the median of \( n \) ordered values is \((n+1)/2\).
- The mode: The most frequently occurring data value.

26. Measures of dispersion tend to measure the spread of a set of data. Common measures of dispersion are:

- The range: The difference between the largest and smallest value.
- The standard deviation: One general formula for the standard deviation of \( n \) values \((x_1, x_2, x_3, \ldots, x_n)\) having a mean \((\bar{x})\) is:

\[
\sqrt{(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + \ldots + (x_n - \bar{x})^2}
\]

27. Write a program that will compute the mean and standard deviation of four input values. Use separate functions to compute the two statistics.

Sample I/O:
Enter 4 VALUES (0 0 0 to quit): 90 75 83 62
AVERAGE: 75.00
STANDARD DEVIATION: 14.45

Enter 4 VALUES (0 0 0 to quit): 58 98 72 72
AVERAGE: 75.00
STANDARD DEVIATION: 14.45

28. Write a program that does Roman numeral arithmetic. Two Roman values and an operator \((+, -, *, /)\) are input. After each Roman operand is input, it is converted to, and redisplayed as, a decimal value. After the operator is input, both the Roman and decimal results of the operation are displayed. Test for, and reject, invalid input data.
29. Write a program to list all the factors of an input integer. If the input value is a prime number, display a message to that effect. Loop for multiple inputs. Exit from the loop if the input value is less than or equal to 1.

30. Write a program that prompts the user for four specifications (column value of upper left corner, row value of upper left corner, width, and height) for a rectangular box, composed of asterisks, to be displayed on the screen. Use a function that accepts the four specifications as parameters.

31. Write a function that tells whether its integer parameter is odd.

32. (Challenge) What common function presented in calculus courses is computed by this C++ function?

```cpp
int f(float n)
{
    if (n < 2)
        return 0;
    else
        if (n == 2)
            return 1;
        else
            return f(n / 2) + 1;
}
```

33. Write a program containing a Boolean function to accept a string and to test whether or not an uppercase letter has been input. Include a loop to allow for multiple inputs. Exit the loop on any non-letter input.

34. Use a `float` function to return the greater of two input values. Include a loop to allow for multiple sets of input. Exit from the loop when the two input values are equal.

35. Write a program that accepts a person’s full name and prints the initials. Include a function that returns the first letter after a space in a string, starting at a given character position.

36. Write a string editor—a program that allows the user to edit a source string of not more than 60 characters. The program is to allow at least the following choices:
   1. Enter a new string
   2. Insert a substring
   3. Replace one substring with another
   4. Delete a substring
   5. Quit

Use a separate function for each of the first four menu options.

37. Write a function `pos (substring,string)` that returns the position of the first character of `substring` in `string`, or returns –1 if `string` does not contain `substring`.

38. Write a recursive function that returns the product of two positive-integer arguments. The function may not use the ‘*’ operator or any loop statement.

39. Write a recursive function that returns the highest power of 2 that is less than or equal to an input number.

   Sample I/O:
   Enter a number: 14
   8
   Enter a number: 233
   128
   Enter a number: 64
   64

40. Write two Boolean functions that each take two Boolean arguments and simulate a logic gate, one for the XOR gate, one for the NAND gate, as these are defined in Chapter 1, Keil/Johnson.

41. Write a function that accepts two strings and an integer. The integer should represent a position in the first string. The function should insert the second string into the first at the position indicated. For example, if string variable `s1` has the value “resion”, a function call `insert(s1,"cur",2)` should result in `s1` having a value “recursion”.

42. Write a function that accepts a string and a character as parameters and returns the number of occurrences of the character in the string. For example, on parameters of “tree” and ‘e’, your function should return 2.

43. Write and demonstrate three functions all named `prompt_for` that accept a prompt message string and return an integer, a character, and a floating-point value, respectively. Test all three functions in one program. What is the C++ feature that lets you use `prompt_for` as the name of all three functions in the same program?

44. Write a function to return the integer value of a string parameter. For example, if the string parameter is “43”, the function should return the integer 43.
Answers to study questions on subprograms

Multiple-choice or T/F:

1. f. The function name is in the header.
2. f. The user's input is rarely known by the programmer; the void type is used for functions that return no value.
3. f. The function declaration introduces it into a source file; it is the definition that spells out what statements are in it.
4. t. Defining functions in order to modularize a program is called procedural abstraction.
5. b. A function must have one definition and may have many declarations.
6. d. Writing function definitions is a way to break down a program or problem into smaller, easily understood, independent units.
7. a. The solution to each subproblem becomes a module or function.
8. f. A goal of modularity is cohesion within each function: the purpose of the function should be singular and focused.
9. d. A prototype, or declaration, is the same as the function header. Functions are often declared at the top of a program and defined after main.
10. c. Breaking down a program into components and grouping related, repeated sequences of statements are the main goals of writing functions.
11. b. Local variables are preferred to global ones because they guarantee against side effects.
12. c. A local variable is allocated when a function begins to execute and disappears when the function terminates.
13. b. The function header has the same form as a function declaration, without the semicolon; the body is a compound statement.
14. b. A reference parameter is specified with "&" after the type name.
15. t. The stack allows a function to share data and to allocate space that is freed up when the function terminates.
16. f. The formal parameter appears in the function definition; the function call may contain actual parameters.
17. b. A value parameter is an expression that is evaluated before the value is passed.
18. c. The return value follows the keyword return and its type should be named in the function header.
19. c. Reference parameters will work; value parameters cannot pass data back, and return values only pass back a single value.
20. d. Actual parameters occur in function calls, formal parameters in function definitions and declarations.
21. c. The ampersand is used to declare a reference parameter, which passes the address of a variable.
22. d. Members of a class are useful, but should represent relatively long-lasting values. Parameters are better.
23. d. Parameters declared in a function's header disappear when that function call's activation record disappears from the stack.
24. a. The stack, like a bulletin board, allows functions to share information as coworkers do.
25. f. Function names may be overloaded. The same name may be used for several functions or constructors with parameter lists of different lengths and parameters of different data types.
26. f. Reference and array parameters have only their addresses copied.
27. b. A data type is associated with every function.
28. t. The branch should allow the function to process the base case, allowing the function to terminate, or the recursive case, causing the function to call itself.
29. t. A recursive function calls itself and must also have a way to terminate without calling itself.
30. a. Java has no repeat statement.
31. a. A function that invokes itself is recursive and the result is a loop.
32. f. A function call may be an expression or statement and needs no type name.
33. t. A function prototype must begin with a type name.
34. 1. A function definition’s header must begin with the function’s return type.

**Short-answer:**
1. return value; reference parameter
2. int sum(float a, float b);

3. void draw_fig(int a, int b);
4. (a) main; (b) find_costs, find_profit; (c) no; (d) no; (e) no; (f) yes
**Study questions on Topic: Database**

1. What is a database table?
2. What is required for an efficient search of a database?
3. What is database filtering?
4. What is a way in MS Excel to find the correlation among values of fields in a database table?
5. Distinguish a pivot table from a database relation or table.
6. What do columns in a database table represent?
7. What do rows represent in a database table?
8. In a pivot table, where do the independent variables go? The dependent variables?
9. What is a database query?
10. Contrast selection queries with projection queries.
11. Give a case where a database table may represent a relationship between two entities.
12. Referring to slide 10 (Entity-Relationship Design), design a database to represent three entities, each entity represented by one table with a primary key. The entities are: customers, products, transaction detail. Detail is one instance of a product purchased, possibly along with other products. Describe the relationships among these entities. Create a small Excel table for each entity.
13. Following the pattern in the previous question, where the third entity represents a joining of one instance of each of the other two entities, design three-table databases for the following:
   (a) job applicants being hired for jobs, so that a sheet can be generated with information about all the new hires for a given period of time;
   (b) poems being published in journal issues, so that a sheet can be generated listing all the poems published for a given journal issue;
   (c) attendees signing up for workshops at a research conference, so that a report can be generated listing all the attendees at a given workshop;
   (d) home buyers visiting open houses, so that a list can be generated of all the home buyers who visited a given house;
   (e) customers buying CDs at music stores, so that a sales slip can be generated;
   (f) CD vendors selling shipments of music to music stores, so that a vendor can list all the shipments made to a certain music store.
14. With copies of the table in listing.xls, perform filter operations to
   (a) select all listings for electric-heated homes in Framingham and Natick with 3 bedrooms or fewer;
   (b) show how many homes there are with each number of bathrooms available;
   (c) show average cost of Wellesley homes;
   (d) show cost of the most expensive listing in Wayland
   (e) select all listings for colonial-style homes in Framingham.
15. Create pivot tables to do the following, with reference to listing.xls:
   (a) show the effect of number of bedrooms and bathrooms on maximum cost
   (b) show the the average costs of houses, by number of bedrooms and number of acres of land.
   (c) show the numbers of houses that are listed, by town and heating type
   (d) show the average costs of houses, by architectural style and town
   (e) show the effect of the lot size and number of baths on minimum cost
   (f) Find total values of oil-heated listings in Natick under $300,000.
16. What important feature of good database design may listing.xls not appear to have? Give arguments that it has the feature and arguments that it doesn’t.
17. Why are there:
   (a) database management systems
   (b) database design principles
   (c) primary keys

**Short answer**

**Multiple choice**

1. To assure that all records are different, database designers use (a) a sort field; (b) queries;
   (c) data analysis (d) searches; (e) a primary key
2. In a pivot table, (a) independent variables are correlated with each other; (b) dependent
variables are correlated; (c) the effect of independent variables on dependent ones is shown; (d) the effect of dependent variables on independent ones is shown; (d) none of these

3. A database normally consists of (a) pixels; (b) tables; (c) keys; (d) protocols; (e) none of these

4. A selection query corresponds to (a) a table; (b) a view; (c) a logical assertion; (d) a set of records; (e) all of these

5. Which of the following is not associated with database management? (a) query design; (b) table design; (c) global control of formatting; (d) entities and relationships; (e) all of these are associated

6. In databases, an object or instance corresponds to a (a) record; (b) table; (c) bit; (d) relation; (e) all of these

1. Write a short C or Java program that displays a table of the first 20 powers of 2 ($2^1, 2^2, \ldots, 2^{20}$), formatted below using bitwise operators (not the pow function) to calculate the entries in the right-hand column of the table. The first three rows will be:

<table>
<thead>
<tr>
<th>n</th>
<th>2 to the nth power</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
</tr>
</tbody>
</table>

2. Write a program that prompts for a person’s first name, up to 80 characters, and display it backwards.

7. In databases, an entity or class of objects is implemented by a (a) record; (b) table; (c) bit; (d) relation; (e) all of these

8. To display information from a database, we use a (a) format command; (b) named style; (c) master page; (d) query; (e) all of these

9. Non-duplication of data in tables is enforced by use of (a) formulas; (b) primary keys; (c) formats; (d) protocols; (e) all of these

10. Write a function that takes a string parameter and swaps its first and second words, using loops to iterate. Define words as sequences of characters separated by a space.

   Text of main:
   ```
   char s[] = “Jane Smith”;
   swap_strings(s);
   cout << s << endl;
   ``

   Output:
   Smith Jane

4. Write a program with no input that for $n = 1$ to 100, displays $n^2(n + 1)^2$ and $\sum_{k=1}^{n} k^3$ and the difference between them.

   Output:
   ```
   n  quotient  sum  diff
   1   1   1   0
   2   ...  ...  ... 
   ... 
   ```
Homework 2 (Structure types and classes)

Graded:
1. Define in your project source code a structure type or class to implement the abstract data type of a web-page object. Instead of “dates visited,” for this version use only one date (a string). In main, declare an instance of this type, and pass it as a parameter (possibly a reference parameter) to the functions that implement the menu choices of adding a web page and displaying a web page. Complete the definitions of these functions. Submit tests of the resulting program. Specs, process, and design documentation need not be resubmitted.

2. Validate the date; i.e.,
   - check to see if there are exactly two slashes;
   - see that some value from 1 to 12 is before the first slash;
   - …1 to 31 between slashes;
   - 2-digit year is in third.

Ungraded:
3. (a) Declare a structure type, with appropriate members, that may be used to declare variables to represent customers. A customer has a customer ID, a name, address, and amount of last purchase. Declare associated global functions, with appropriate parameters, to input and display member data. (You need not define these functions.)
   (b) Define the equivalent class, with member function declarations. (No function definitions needed.)

4. See the program coursereg.cpp in the CS II network directory and on the slide, “A course-registration class.” Add to it a declaration and definition of a Boolean member function, is_null(), that tells whether the student ID or course ID member of a registration object is 0, returning true (1), if so, otherwise returning false (0). Modify main to use a loop to input and display course-registration objects until the user enters a 0 and is_null thus returns true.

Sample I/O:
---Type 0 to exit---
Student ID, course ID: 124 63152
124 63152
Student ID, course ID: 329 63252
329 63252
Student ID, course ID: 782 43320
782 43320
Student ID, course ID: 0 0
Homework 3 (Arrays)

Graded:

1. In your semester project, implement a *collection* of web-page records as follows and submit test results, documenting your changes. Note that modularity is required; your collection must pass between functions as a class member or parameter, not as a global variable.
   - Declare a structure type or class that has two members: an array of instances of your web-page structure, and an integer representing the number of elements of this array that are currently in use.
   - Declare an instance of this type in your `main` in place of the instance of the web-page type.
   - If you use a structure type, pass this `collection` object as a value or reference parameter to each of the functions called from `main`. If you use a class, the collection will be accessible to all member functions.
   - Define two of these functions properly: the one that inputs one record and adds it to the collection, and the one that displays all records.
   - *Suggestion:* in the functions that take the collection as a parameter, call the functions that display and input one web site object.

Ungraded:

2. Write a program that prompts for a series of integers, then stores them in an array, then
   (a) displays the number of 13’s found among the input values;
   (b) tells whether the array is in ascending order.

3. Write a C or Java program that uses a two-dimensional array, initialized as below, to calculate and display sales summaries by quarter and by region for a company with these sales, in millions of dollars.

<table>
<thead>
<tr>
<th>Atlantic</th>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Q4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Midwest</td>
<td>82</td>
<td>78</td>
<td>92</td>
<td>74</td>
</tr>
<tr>
<td>West</td>
<td>157</td>
<td>183</td>
<td>256</td>
<td>173</td>
</tr>
</tbody>
</table>

   A table like the following should be displayed. The *Year* column and the *All regions* row should be calculated by the program, not initialized by the programmer. It’s easiest if loops are used.

<p>| (Sales in millions of dollars) |</p>
<table>
<thead>
<tr>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Q4</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atlantic</td>
<td>34</td>
<td>21</td>
<td>29</td>
<td>42</td>
</tr>
<tr>
<td>Midwest</td>
<td>82</td>
<td>78</td>
<td>92</td>
<td>74</td>
</tr>
<tr>
<td>West</td>
<td>157</td>
<td>183</td>
<td>256</td>
<td>173</td>
</tr>
<tr>
<td>All regions</td>
<td>273</td>
<td>282</td>
<td>377</td>
<td>289</td>
</tr>
</tbody>
</table>

4. Write and test a subprogram that accepts as parameters an unsorted array of integers, and the size of this collection, and returns the second largest value in the array.
Homework 4 (File I/O)

Graded:
1. Extend your project by writing the definitions of the functions that save and retrieve web-site records. Write and test functions to retrieve and save your array-based collection of web-page records for the project.

   • If you use a class, write member functions of the collection class such that function `save` opens a file for output and loops through your array, calling a function `save` of the web-page class once for each element of the collection.

   • The `save` for web page records should take a parameter that is the file stream object. When the collection `save` function has finished the file-writing loop, it should close the file.

   • Write a similar collection function, `retrieve`, that opens a file for input and loops until `eof()` to call a `retrieve` function of the web-page class.

   • Document by explaining data file format.

   • Test by inputting several records, saving and ending the session, then retrieve and display the results. Show a log of save, retrieve, and display operations (you need not show hard copy of input tests).

Ungraded:
2. Write a program to read a set of floating-point values in a file, as below, and display these values and their maximum. The program should read as many or as few values as are in the file.

   Sample output:
   29.3  42.9  56  19.8  62.1  9.0  49.4
   Average: 38.357

3. Write a program that reads a text file named by the user, line by line, and copies it to another text file, named by the user, which is a copy of the original input file except that each line begins with its line number followed by a colon.

   Sample partial output, given user input of "echofil.cpp":
   1: // echofile.cpp
   2: // Reads a text file and displays its contents.
   3: #include <iostream.h>
   4: #include <fstream.h>
   5: void main()