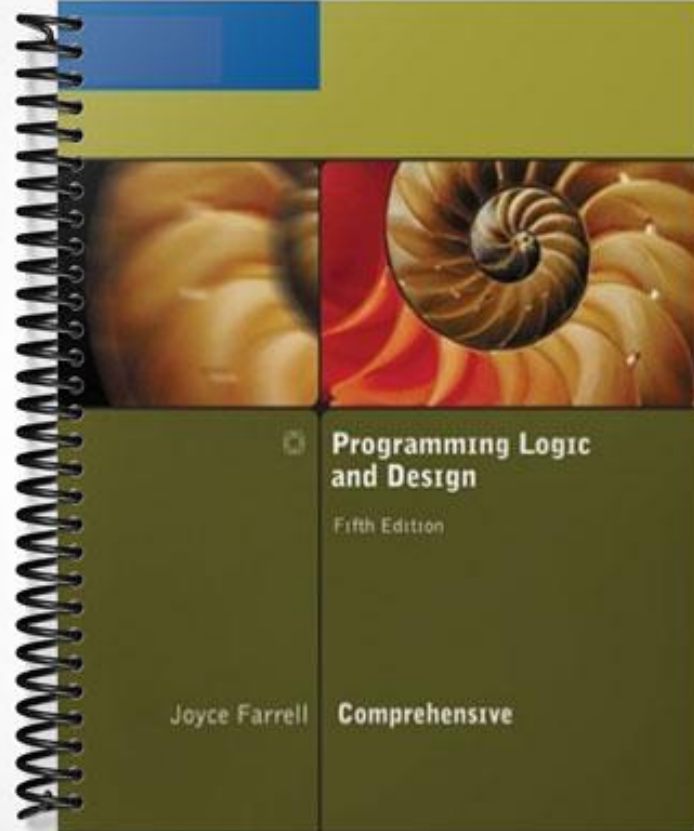


TEST BANK



Programming Logic
and Design

Fifth Edition

Joyce Farrell

Comprehensive

ch02

True/False

Indicate whether the statement is true or false.

- ___ 1. With a selection structure, you perform an action or task, and then you perform the next action, in order.
- ___ 2. In a structured program, any structure can be nested within another structure.
- ___ 3. When you encounter an eof question in a flowchart, you know that either a selection or loop structure should begin.
- ___ 4. When you write a series of decisions using the case structure, the computer still makes a series of individual decisions.
- ___ 5. When you use a while loop, at least one performance of the action inside the loop body always occurs.

Multiple Choice

Identify the choice that best completes the statement or answers the question.

- ___ 6. The following pseudocode is an example of a(n) ___ structure:
 get firstNumber
 get secondNumber
 add firstNumber and secondNumber
 print result
 - a. sequence
 - b. decision
 - c. loop
 - d. nested

- ___ 7. The following pseudocode is an example of a(n) ___ structure:
 if firstNumber is bigger than secondNumber then
 print firstNumber
 else
 print secondNumber
 - a. sequence
 - b. decision
 - c. loop
 - d. nested

- ___ 8. Fill in the blank in the following pseudocode:
 if someCondition is true then
 do oneProcess

 ___ do theOtherProcess
 - a. then
 - b. while
 - c. do
 - d. else

- ___ 9. The following pseudocode is an example of a(n) ___ structure:
 get number
 while number is positive
 add to sum
 get number
 - a. sequence
 - c. loop

- b. decision
- d. nested

- ___ 10. Another name for a loop structure is ____.
- a. execution
 - b. selection
 - c. iteration
 - d. case

- ___ 11. The following pseudocode is an example of ____.
- ```
do stepA
do stepB
if conditionC is true then
 do stepD
else
 do stepE
endif
while conditionF is true
 do stepG
endwhile
```
- a. nesting
  - b. stacking
  - c. single alternative structures
  - d. a posttest

- \_\_\_ 12. The following pseudocode is an example of \_\_\_\_.
- ```
if conditionA is true then
  do stepE
else
  do stepB
  do stepC
  do stepD
endif
```
- a. nesting
 - b. stacking
 - c. a posttest
 - d. a pretest

- ___ 13. The maximum number of entry points that any programming structure can have is ____.
- a. zero
 - b. one
 - c. three
 - d. five

- ___ 14. The statement that best describes the following pseudocode is: ____.
- ```
if conditionA is true then
 do stepE
else
 do stepB
 if conditionF is true then
 while conditionI is true
 do stepJ
 endwhile
 else
 do stepG
 endif
do stepD
endif
```
- a. A decision is nested in a sequence
  - b. A decision is nested in a loop
  - c. A loop is nested in a decision that is nested in another decision
  - d. A sequence is nested inside a decision nested in another decision

- \_\_\_ 15. Suppose a program will read 100 data records, and you read the first data record in a statement that is separate from the other 99. This is called a \_\_\_ read.
- nested
  - stacked
  - posttest
  - priming
- \_\_\_ 16. The following pseudocode reads a number from the user, multiplies it by 2 and prints the result. The \_\_\_ program statement should replace the ? to make this program functional and structured.
- ```

get inputNumber
while not eof
    calculatedAnswer = inputNumber * 2
    print calculatedAnswer
    ?
endwhile

```
- no statement is needed
 - if done then exit
 - get inputNumber
 - print inputNumber
- ___ 17. Years ago, programmers could avoid using structure by inserting a “___” statement into their pseudocode.
- loop
 - go next
 - next
 - go to
- ___ 18. Structured programs can be easily broken down into routines or ___ that can be assigned to any number of programmers.
- segments
 - modules
 - units
 - sequences
- ___ 19. One way to straighten out a flowchart segment that is not structured is to use what is called the “___” method.
- spaghetti code
 - spaghetti bowl
 - restructuring
 - priming
- ___ 20. The following pseudocode might be rewritten using a(n) ___ structure:
- ```

if class = "Freshman" then
 tuitionFee = 75
else
 if class = "Sophomore" then
 tuitionFee = 50
 else
 if class = "Junior" then
 tuitionFee = 30
 else
 tuitionFee = 10
 endif
 endif
endif

```
- if-then-else
  - case
  - while
  - do while
- \_\_\_ 21. \_\_\_ is considered to be a convenience structure.
- if-then-else
  - while
  - case
  - sequence



- \_\_\_ 34. basic unit of programming logic
- \_\_\_ 35. branch of a decision where nothing is done
- \_\_\_ 36. continue to repeat actions based on the answer to a question
- \_\_\_ 37. attaching structures end-to-end
- \_\_\_ 38. statement that reads the first input data record
- \_\_\_ 39. used when there are several distinct possible values for a single variable you are testing, and each value requires a different course of action
- \_\_\_ 40. `do-while` and `do-until`

### Short Answer

- 41. Define the term structure as it relates to programming.
- 42. Describe a loop structure.
- 43. All logic problems can be solved using only these three structures—sequence, selection, and loop. The three structures, of course, can be combined in an infinite number of ways. What are two general ways structures can be combined?
- 44. What are the characteristics of a structured program?
- 45. Explain the difference between the representation of a decision structure and a loop in a flowchart.
- 46. Why is it best to use only three programming structures?
- 47. What is the purpose of the case structure?
- 48. What is the difference between a `while` and a `do-while` or `do-until` loop?
- 49. Describe the difference between a pretest and a posttest loop.
- 50. Rewrite the following as a `while` loop:  
do  
    pay bills  
while more bills remain to be paid

## ch02

### Answer Section

#### TRUE/FALSE

- |           |        |         |
|-----------|--------|---------|
| 1. ANS: F | PTS: 1 | REF: 46 |
| 2. ANS: T | PTS: 1 | REF: 53 |
| 3. ANS: T | PTS: 1 | REF: 58 |
| 4. ANS: T | PTS: 1 | REF: 70 |
| 5. ANS: F | PTS: 1 | REF: 73 |

#### MULTIPLE CHOICE

- |            |        |              |
|------------|--------|--------------|
| 6. ANS: A  | PTS: 1 | REF: 46      |
| 7. ANS: B  | PTS: 1 | REF: 46      |
| 8. ANS: D  | PTS: 1 | REF: 47      |
| 9. ANS: C  | PTS: 1 | REF: 48      |
| 10. ANS: C | PTS: 1 | REF: 48      |
| 11. ANS: B | PTS: 1 | REF: 48   49 |
| 12. ANS: A | PTS: 1 | REF: 50      |
| 13. ANS: B | PTS: 1 | REF: 52      |
| 14. ANS: C | PTS: 1 | REF: 52      |
| 15. ANS: D | PTS: 1 | REF: 54      |
| 16. ANS: C | PTS: 1 | REF: 58      |
| 17. ANS: D | PTS: 1 | REF: 58      |
| 18. ANS: B | PTS: 1 | REF: 60      |
| 19. ANS: B | PTS: 1 | REF: 64   65 |
| 20. ANS: B | PTS: 1 | REF: 69   70 |
| 21. ANS: C | PTS: 1 | REF: 70      |
| 22. ANS: C | PTS: 1 | REF: 70      |
| 23. ANS: A | PTS: 1 | REF: 70      |
| 24. ANS: D | PTS: 1 | REF: 71      |
| 25. ANS: A | PTS: 1 | REF: 71      |
| 26. ANS: C | PTS: 1 | REF: 71      |

#### COMPLETION

- |                       |        |         |
|-----------------------|--------|---------|
| 27. ANS: sequence     |        |         |
|                       | PTS: 1 | REF: 46 |
| 28. ANS: if-then-else |        |         |
|                       | PTS: 1 | REF: 47 |
| 29. ANS: block        |        |         |

PTS: 1                    REF: 50  
30. ANS:  
loop-controlling question  
loop controlling question

PTS: 1                    REF: 57  
31. ANS: single

PTS: 1                    REF: 70

## **MATCHING**

|            |        |         |
|------------|--------|---------|
| 32. ANS: G | PTS: 1 | REF: 44 |
| 33. ANS: D | PTS: 1 | REF: 46 |
| 34. ANS: A | PTS: 1 | REF: 46 |
| 35. ANS: E | PTS: 1 | REF: 47 |
| 36. ANS: H | PTS: 1 | REF: 48 |
| 37. ANS: F | PTS: 1 | REF: 48 |
| 38. ANS: B | PTS: 1 | REF: 54 |
| 39. ANS: C | PTS: 1 | REF: 68 |
| 40. ANS: I | PTS: 1 | REF: 71 |

## **SHORT ANSWER**

41. ANS:  
In the mid-1960s, mathematicians proved that any program, no matter how complicated, can be constructed using one or more of only three structures. A structure is a basic unit of programming logic; each structure is a sequence, selection, or loop. With these three structures alone, you can diagram any task, from doubling a number to performing brain surgery. You can diagram each structure with a specific configuration of flowchart symbols.

PTS: 1                    REF: 46

42. ANS:  
In a loop structure, you continue to repeat actions based on the answer to a question. In the most common type of loop, you first ask a question; if the answer requires an action, you perform the action and ask the original question again. If the answer requires that the action be taken again, you take the action and then ask the original question again. This continues until the answer to the question is such that the action is no longer required; then you exit the structure. You may hear programmers refer to looping as repetition or iteration.

PTS: 1                    REF: 48

43. ANS:  
You can have a sequence of tasks followed by a selection, or a loop followed by a sequence. Attaching structures end-to-end is called stacking structures.



Besides stacking structures, you can replace any individual tasks or steps in a structured flowchart diagram or pseudocode segment with additional structures. In other words, any sequence, selection, or loop can contain other sequences, selections, or loops. For example, you can have a sequence of three tasks on one side of a selection. Placing a structure within another structure is called nesting the structures.

PTS: 1 REF: 48 | 50

44. ANS:

- A structured program includes only combinations of the three basic structures— sequence, selection, and loop. Any structured program might contain one, two, or all three types of structures.
- Structures can be stacked or connected to one another only at their entry or exit points.
- Any structure can be nested within another structure.

PTS: 1 REF: 53

45. ANS:

With a selection structure, the logic goes in one of two directions after the question, and then the flow comes back together; the question is not asked a second time. However, in a loop, if the answer to the question results in the loop being entered and the loop statements executing, then the logic returns to the question that started the loop; when the body of a loop executes, the question that controls the loop is always asked again.

PTS: 1 REF: 55

46. ANS:

- *Clarity*—The number-doubling program is a small program. As programs get bigger, they get more confusing if they're not structured.
- *Professionalism*—All other programmers (and programming teachers you might encounter) expect your programs to be structured. It's the way things are done professionally.
- *Efficiency*—Most newer computer languages are structured languages with syntax that lets you deal efficiently with sequence, selection, and looping. Older languages, such as assembly languages, COBOL, and RPG, were developed before the principles of structured programming were discovered. However, even programs that use those older languages can be written in a structured form, and structured programming is expected on the job today. Newer languages such as C#, C++, and Java enforce structure by their syntax.
- *Maintenance*—You, as well as other programmers, will find it easier to modify and maintain structured programs as changes are required in the future.
- *Modularity*—Structured programs can be easily broken down into routines or modules that can be assigned to any number of programmers. The routines are then pieced back together like modular furniture at each routine's single entry or exit point. Additionally, often a module can be used in multiple programs, saving development time in the new project.

PTS: 1 REF: 60

47. ANS:

You can use the case structure when there are several distinct possible values for a single variable you are testing, and each value requires a different course of action.

When using the case structure, you test a variable against a series of values, taking appropriate action based on the variable's value. To many, such programs seem easier to read, and the case structure is allowed because the same results *could* be achieved with a series of structured selections (thus making the program structured). That is, if the first program is structured and the second one reflects the first one point by point, then the second one must also be structured.

PTS: 1 REF: 68 | 70

48. ANS:

In a `while` loop, you ask a question and, depending on the answer, you might or might not enter the loop to execute the loop's procedure. Conversely, in `do-while` and `do-until` loops, you ensure that the procedure executes at least once; then, depending on the answer to the controlling question, the loop may or may not execute additional times. In a `do-while` loop, the loop body continues to execute as long as the answer to the controlling question is yes, or true. In a `do-until` loop, the loop body continues to execute as long as the answer to the controlling question is no, or false; that is, the body executes *until* the controlling question is yes or true.

PTS: 1 REF: 71

49. ANS:

In a `while` loop, the question that controls a loop comes at the beginning, or "top," of the loop body. A `while` loop is also called a pretest loop because a condition is tested before entering the loop even once. In a `do-while` or `do-until` loop, the question that controls the loop comes at the end, or "bottom," of the loop body. `Do-while` and `do-until` loops are also called posttest loops because a condition is tested after the loop body has executed.

PTS: 1 REF: 71

50. ANS:

```
pay bills
while there are more bills to pay
 pay bills
endwhile
```

PTS: 1 REF: 71 | 72