



Programming Logic and Design

Seventh Edition

Chapter 3

Understanding Structure



Objectives

In this chapter, you will learn about:

- The disadvantages of unstructured spaghetti code
- The three basic structures—sequence, selection, and loop
- Using a priming input to structure a program
- The need for structure
- Recognizing structure
- Structuring and modularizing unstructured logic

The Disadvantages of Unstructured Spaghetti Code

- **Spaghetti code**
 - Logically snarled program statements
 - Often a complicated mess
 - Programs often work but are difficult to read and maintain
 - Confusing and prone to error
- **Unstructured programs**
 - Do not follow the rules of structured logic
- **Structured programs**
 - Follow the rules of structured logic

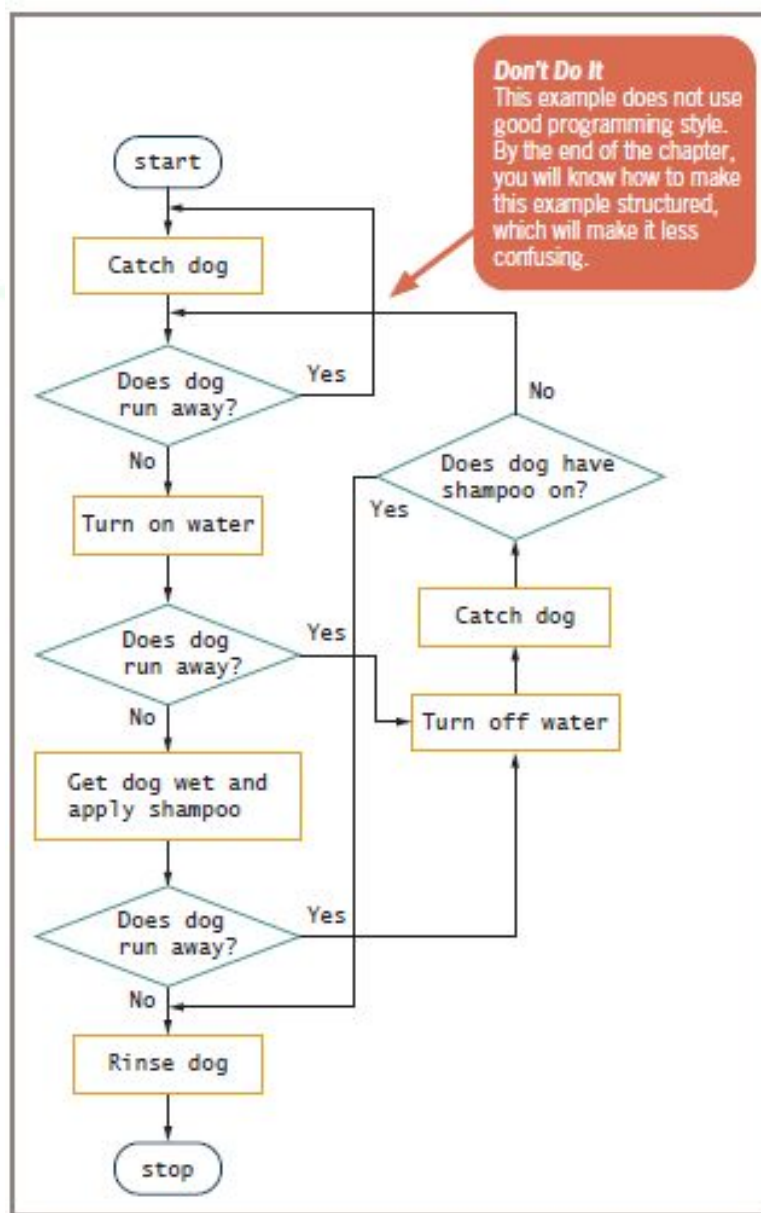


Figure 3-1 Spaghetti code logic for washing a dog

Understanding the Three Basic Structures

- **Structure**
 - Basic unit of programming logic
 - **Sequence structure**
 - Perform actions in order
 - No branching or skipping any task
 - **Selection structure (decision structure)**
 - Ask a question, take one of two actions
 - **Dual-alternative ifs** or **single-alternative ifs**
 - **Loop structure**
 - Repeat actions while a condition remains true

Understanding the Three Basic Structures (continued)

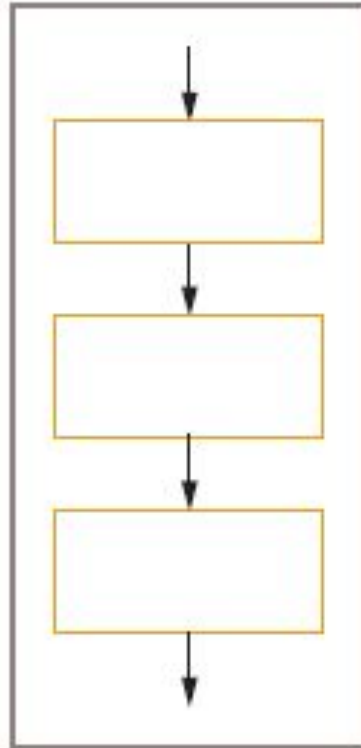


Figure 3-2 Sequence structure

Understanding the Three Basic Structures (continued)

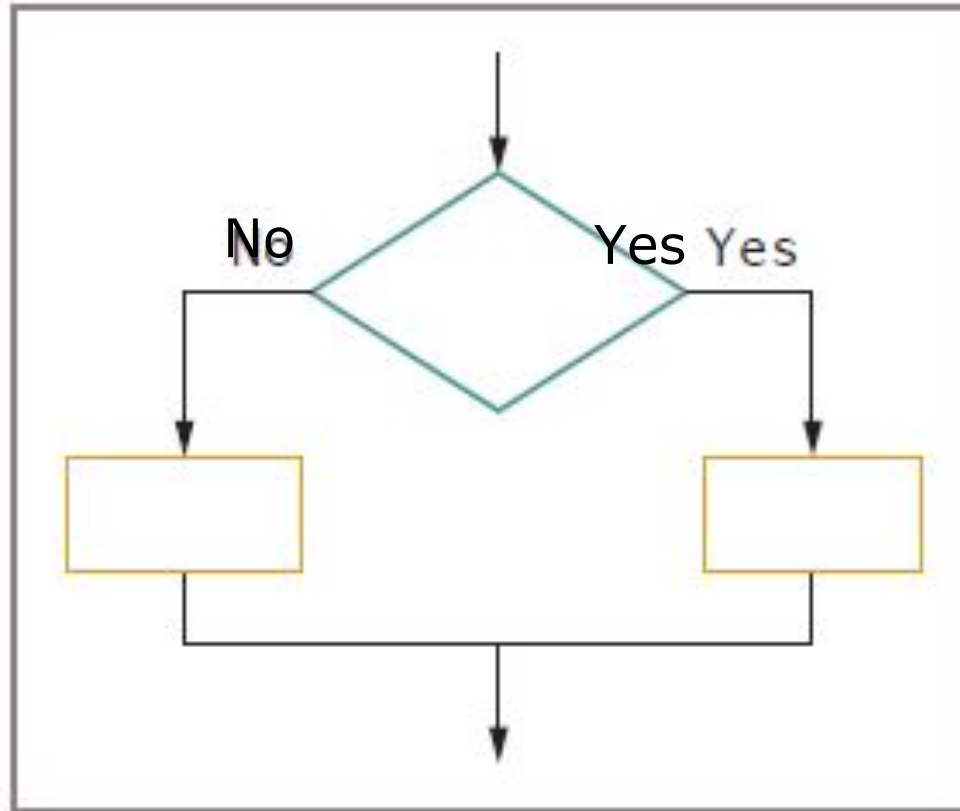


Figure 3-3 Selection structure

Understanding the Three Basic Structures (continued)

- **Dual-alternative ifs**

- Contain two alternatives
- The **if-then-else** structure

```
if someCondition is true then  
    do oneProcess  
  
else  
    do theOtherProcess  
  
endif
```


Understanding the Three Basic Structures (continued)

- **Single-alternative ifs**

```
if employee belongs to dentalPlan then  
    deduct $40 from employeeGrossPay
```

- An `else` clause is not required

- **null case**

- Situation where nothing is done

Understanding the Three Basic Structures (continued)

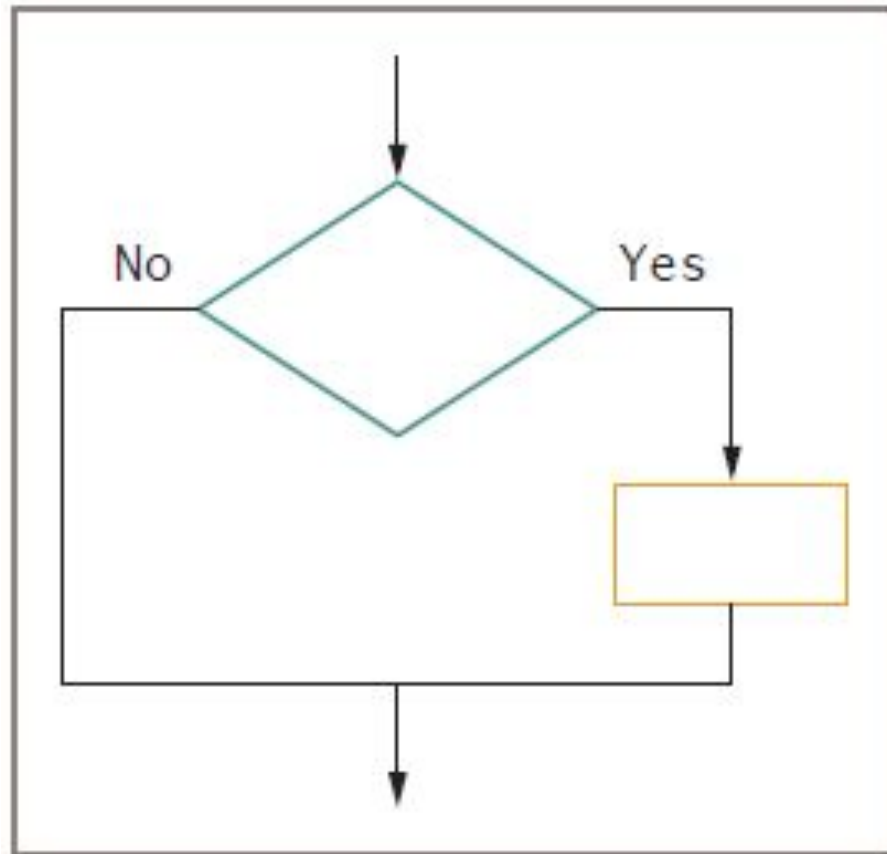


Figure 3-4 Single-alternative selection structure

Understanding the Three Basic Structures (continued)

- **Loop structure**

- Repeats a set of actions while a condition remains true
 - **Loop body**
- Also called **repetition** or **iteration**
- Condition is tested first in the most common form of loop
- The **while...do** or **while loop**

Understanding the Three Basic Structures (continued)

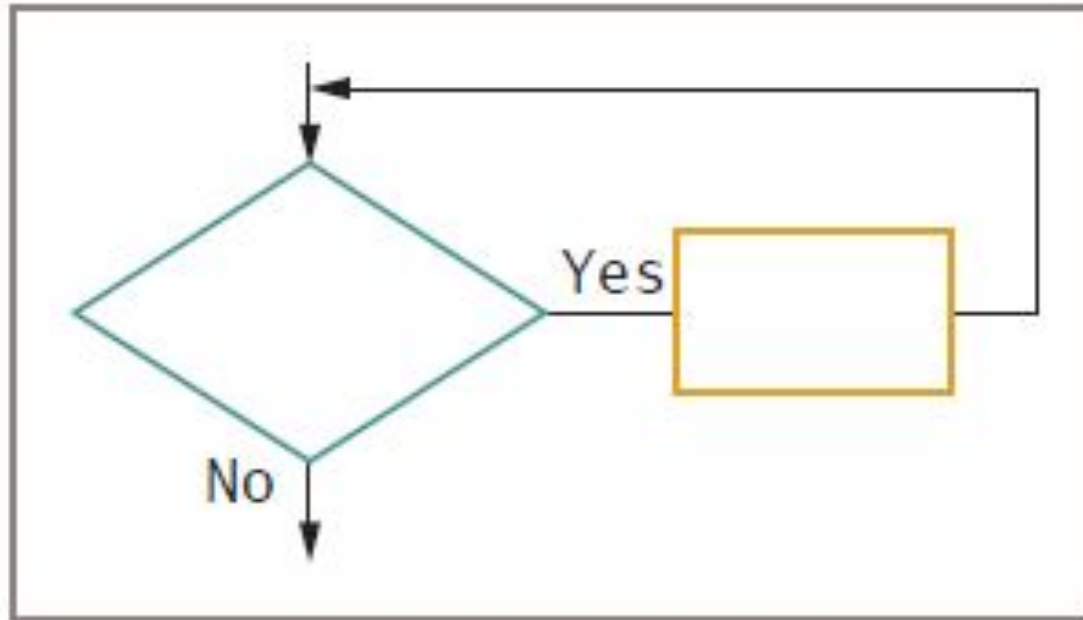


Figure 3-5 Loop structure

Understanding the Three Basic Structures (continued)

- **Loop structure**

```
while testCondition continues to be true  
    do someProcess
```

```
while you continue to be hungry  
    take another bite of food  
    determine if you still feel hungry
```

Understanding the Three Basic Structures (continued)

- All logic problems can be solved using only sequence, selection, and loop
- Structures can be combined in an infinite number of ways
- **Stacking structures**
 - Attaching structures end-to-end
- **End-structure statement**
 - Indicates the end of a structure
 - The `endif` statement ends an `if-then-else` structure
 - The `endwhile` statement ends a loop structure

Understanding the Three Basic Structures (continued)

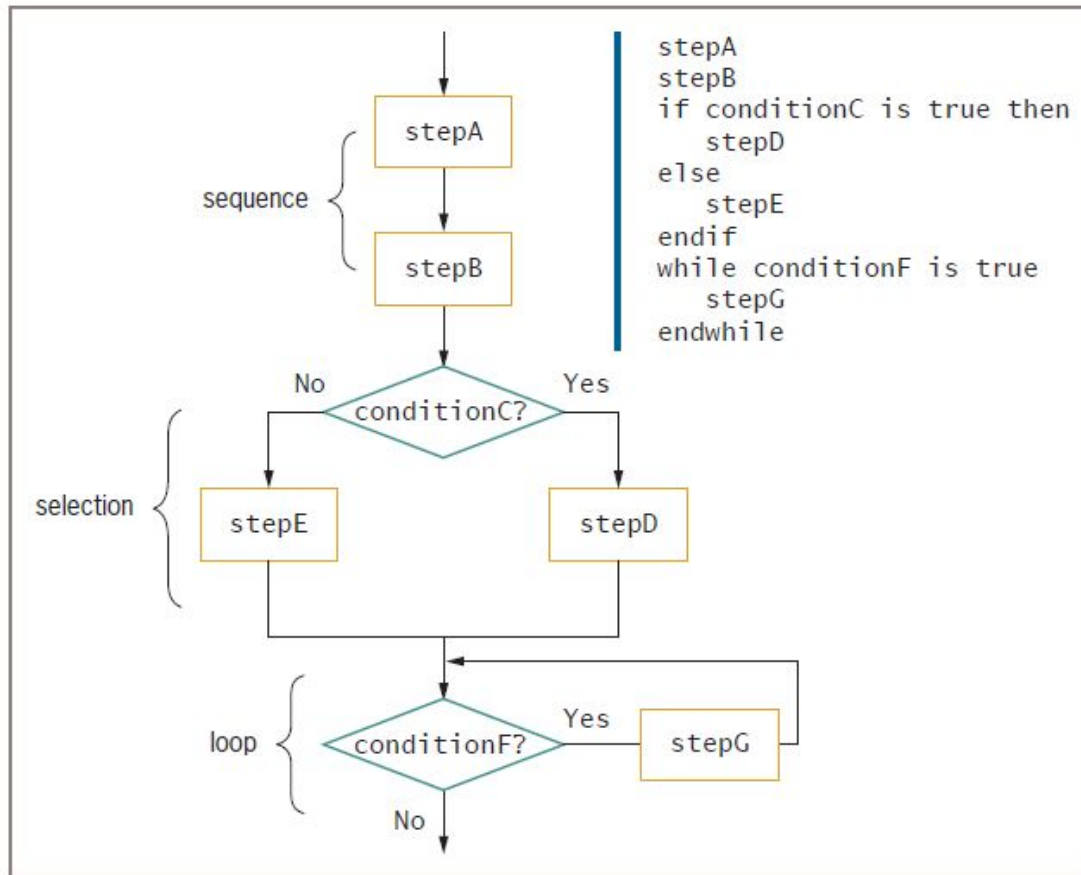


Figure 3-6 Structured flowchart and pseudocode with three stacked structures

Understanding the Three Basic Structures (continued)

- Any individual task or step in a structure can be replaced by a structure
- **Nesting structures**
 - Placing one structure within another
 - Indent the nested structure's statements
- **Block**
 - A group of statements that execute as a single unit

Understanding the Three Basic Structures (continued)

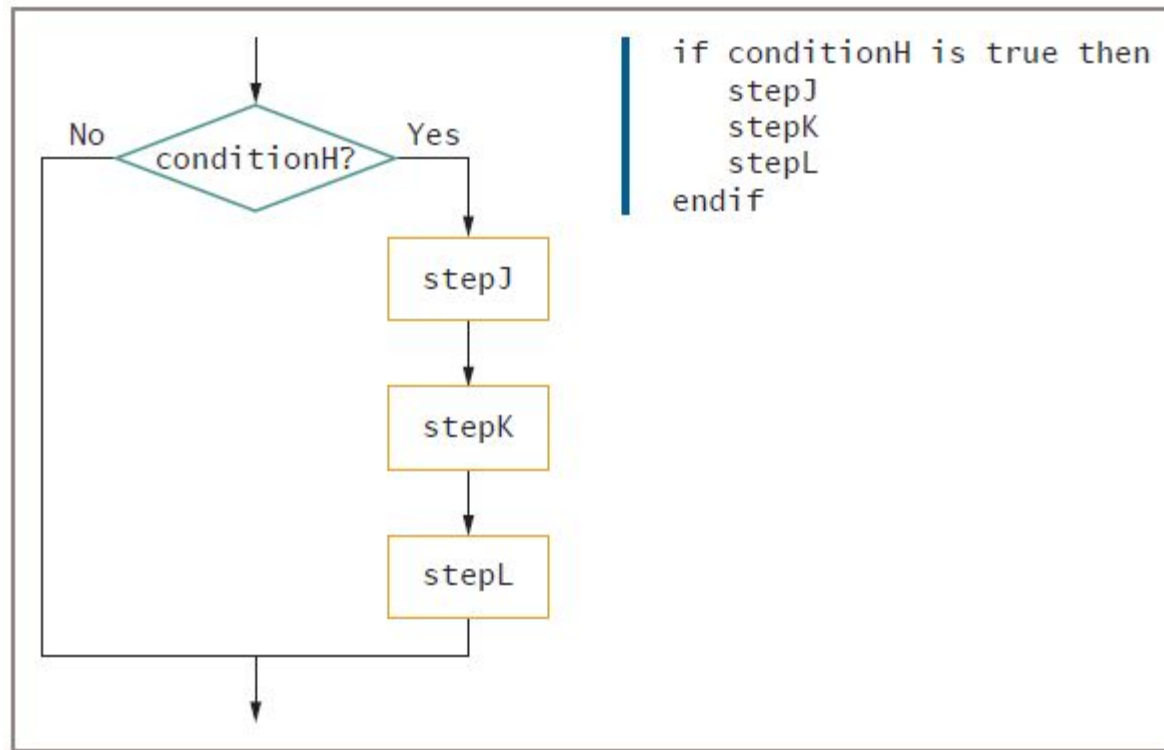


Figure 3-7 Flowchart and pseudocode showing nested structures—
a sequence nested within a selection

Understanding the Three Basic Structures (continued)

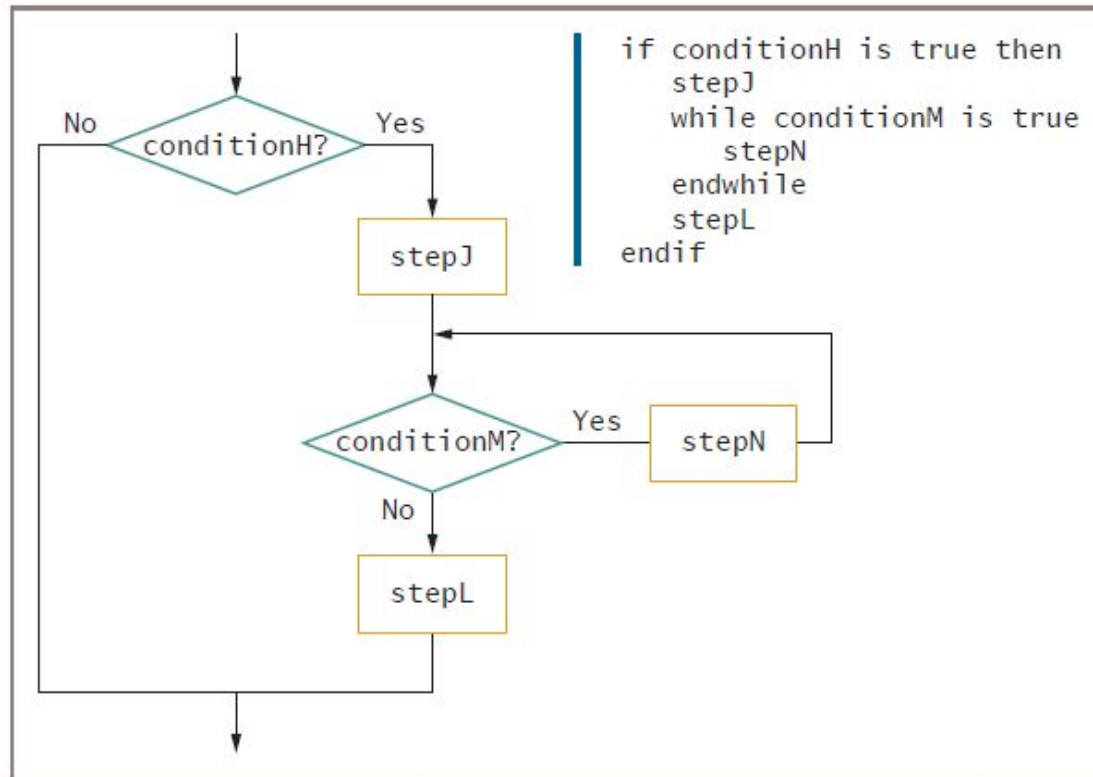


Figure 3-8 Flowchart and pseudocode showing nested structures— a loop nested within a sequence, nested within a selection

Understanding the Three Basic Structures (continued)

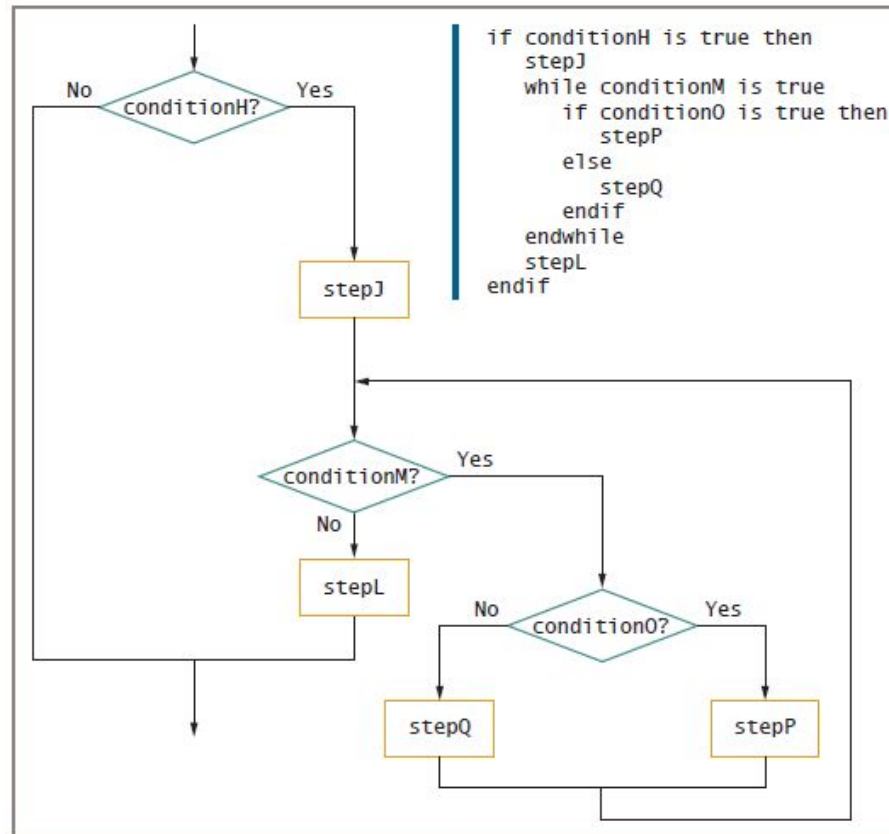


Figure 3-9 Flowchart and pseudocode for a selection within a loop within a sequence within a selection

Understanding the Three Basic Structures (continued)

- Structured programs have the following characteristics:
 - Include only combinations of the three basic structures
 - Each structure has a single entry point and a single exit point
 - Structures can be stacked or connected to one another only at their entry or exit points
 - Any structure can be nested within another structure

Using a Priming Input to Structure a Program

- **Priming input (or priming read)**
 - Reads the first input data record
 - Is outside the loop that reads the rest of the records
 - Helps keep the program structured
- Analyze a flowchart for structure one step at a time
- Watch for unstructured loops that do not follow this order
 - First ask a question
 - Take action based on the answer
 - Return to ask the question again

Using a Priming Input to Structure a Program (continued)

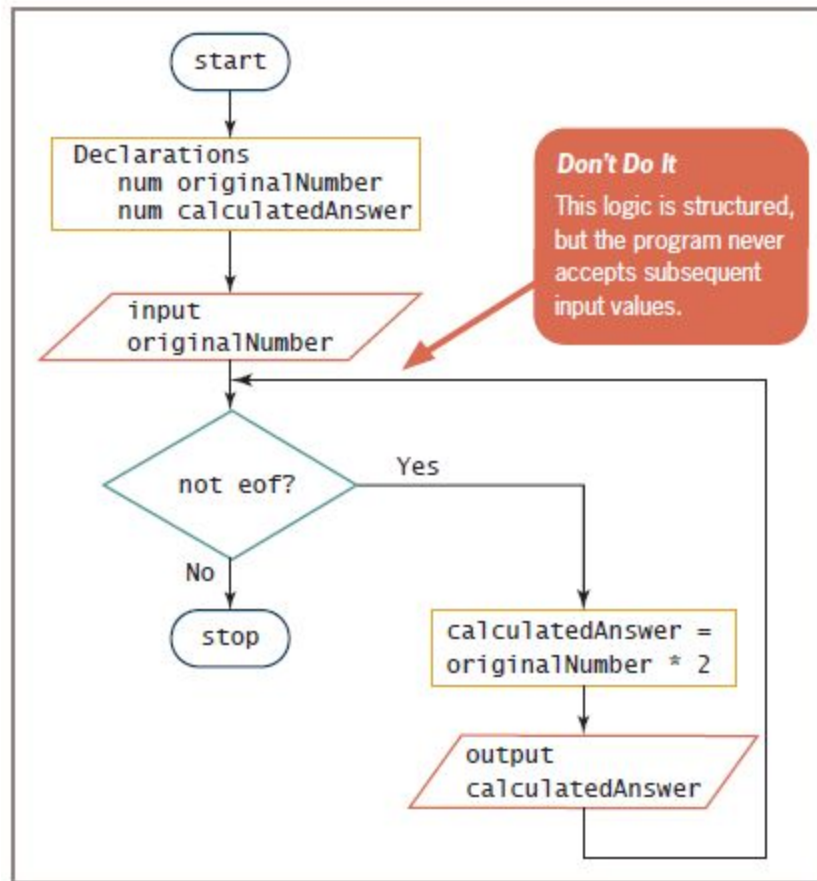


Figure 3-15 Structured, but nonfunctional, flowchart of number-doubling problem

Using a Priming Input to Structure a Program (continued)

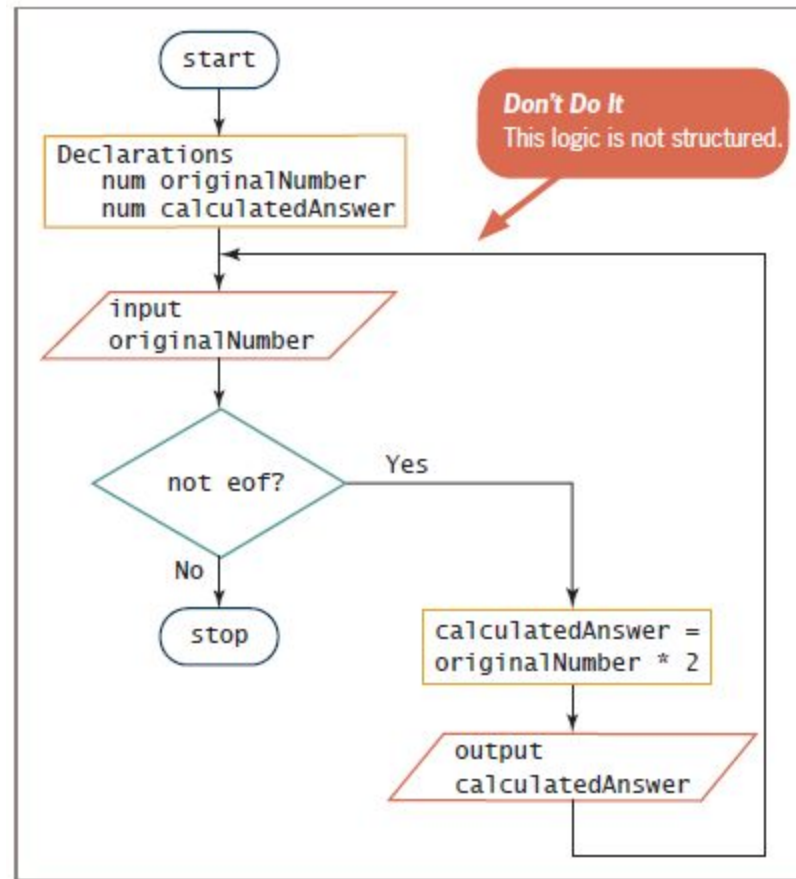


Figure 3-16 Functional but unstructured flowchart

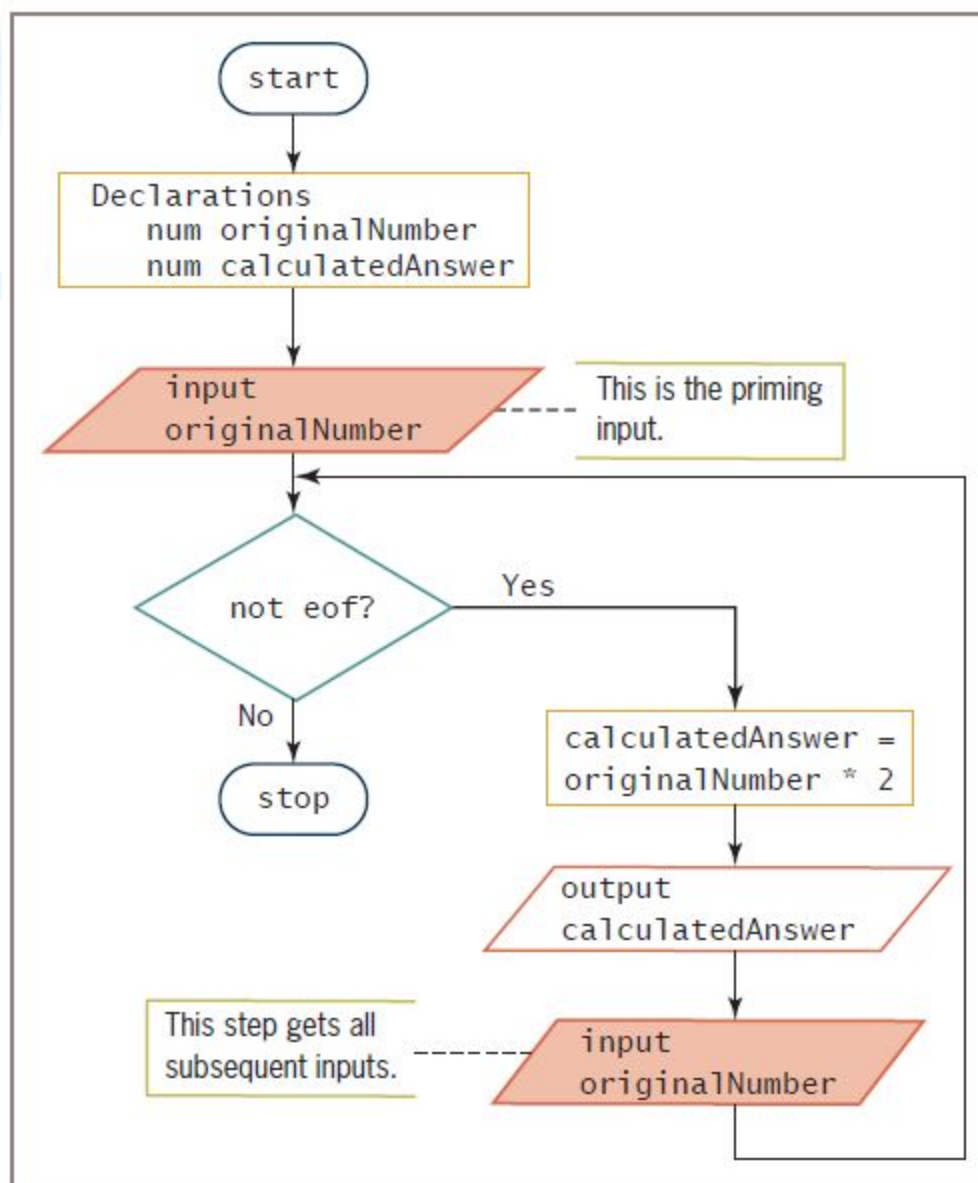


Figure 3-17 Functional, structured flowchart for the number-doubling problem

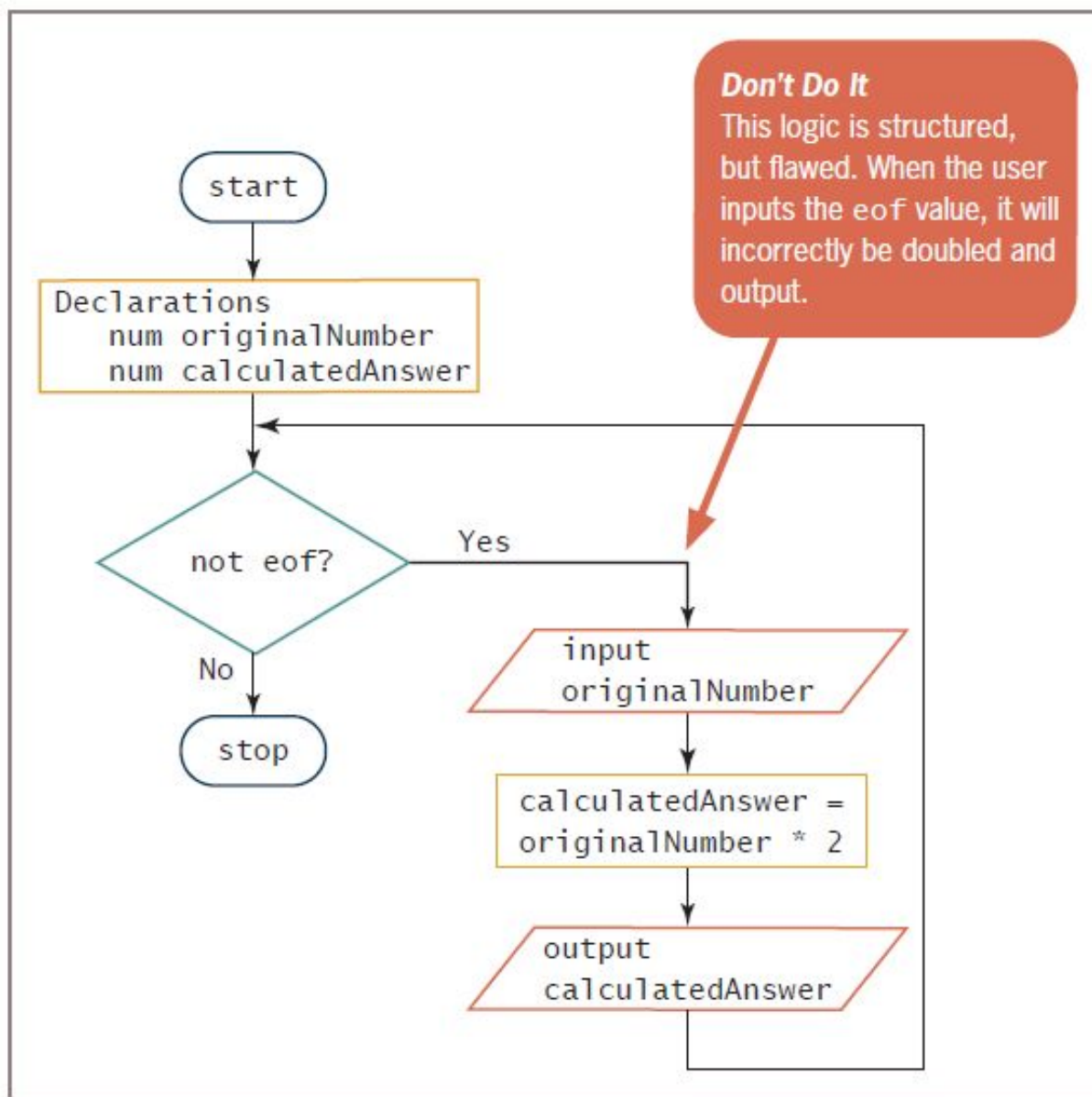


Figure 3-18 Structured but incorrect solution to the number-doubling problem

Understanding the Reasons for Structure

- *Clarity*—unstructured programs are confusing
- *Professionalism*—other programmers expect it
- *Efficiency*—most languages support it
- *Ease of maintenance*—other programmers find it easier to read
- *Supports modularity*—easily broken down into modules
- It can be difficult to detect whether a flowchart is structured

Recognizing Structure

A Structured Flowchart

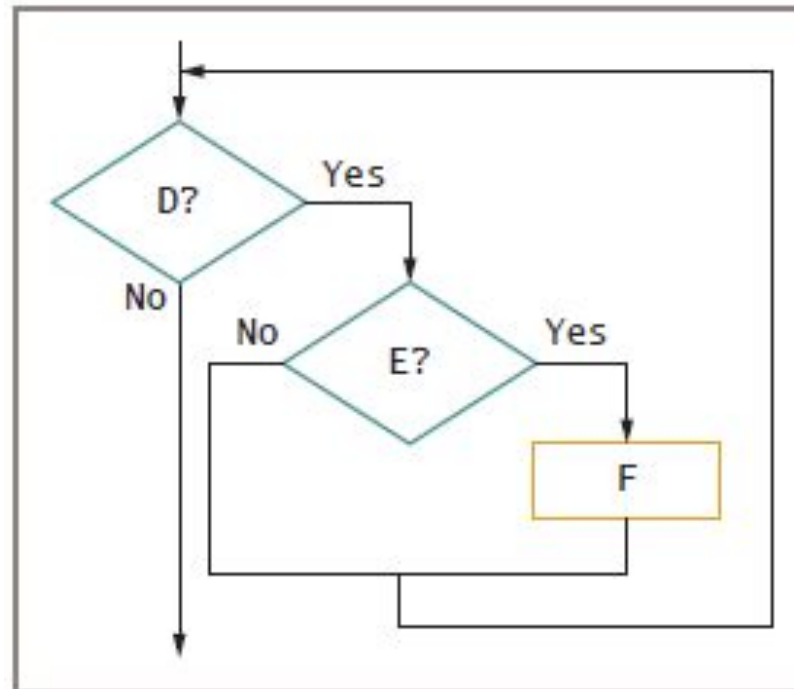


Figure 3-20 Example 2

Recognizing Structure (continued)

An Unstructured Flowchart

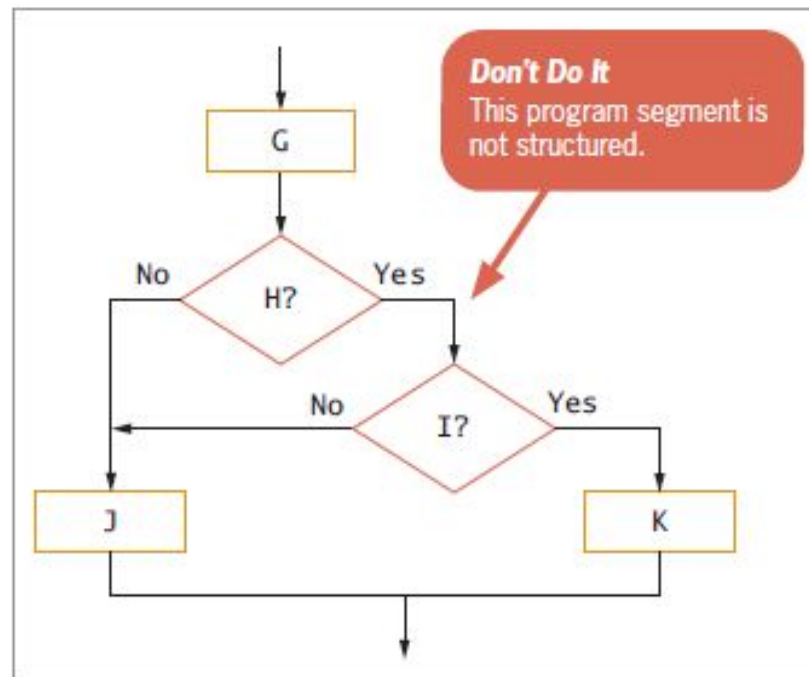


Figure 3-21 Example 3

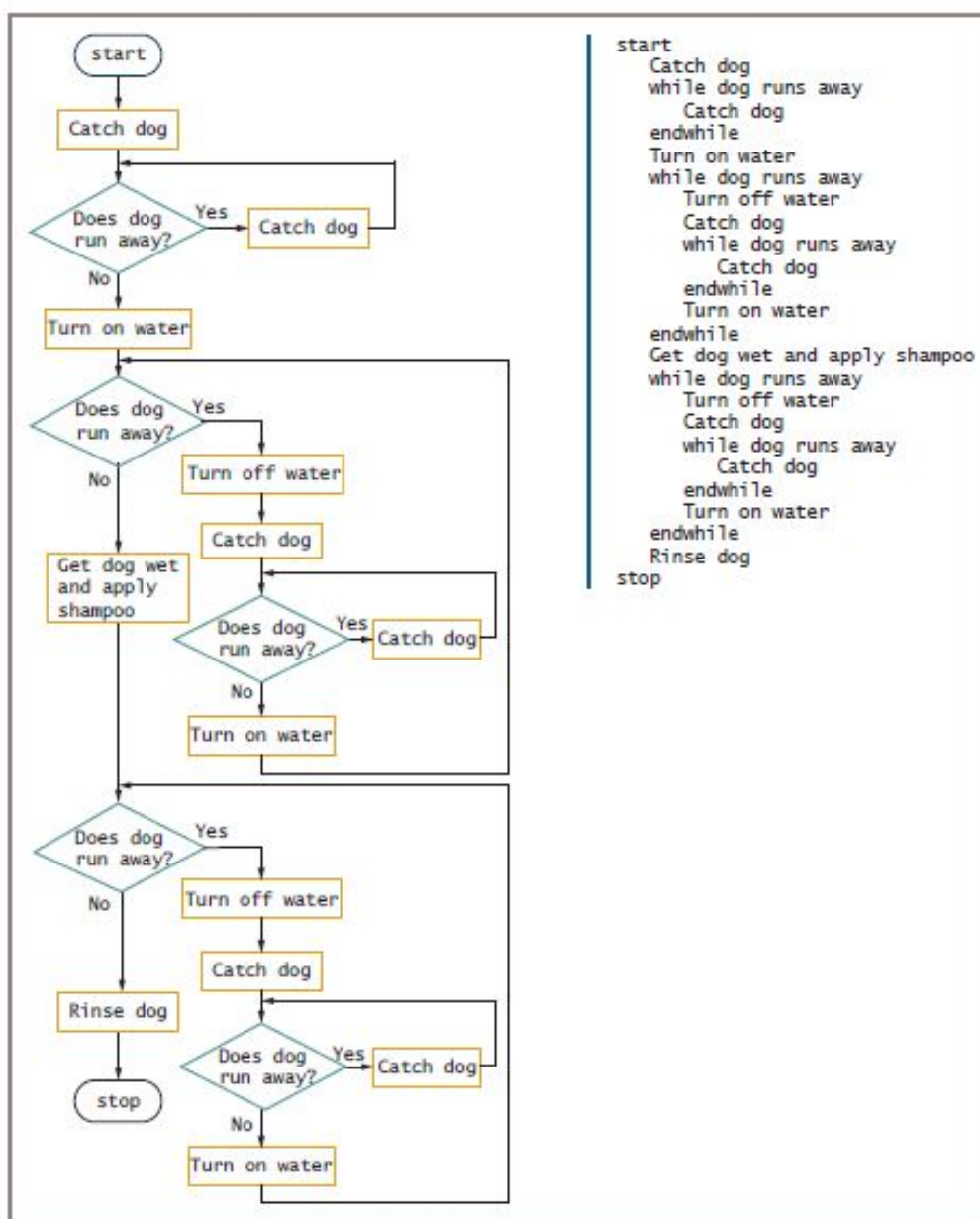


Figure 3-23 Structured dog-washing flowchart and pseudocode

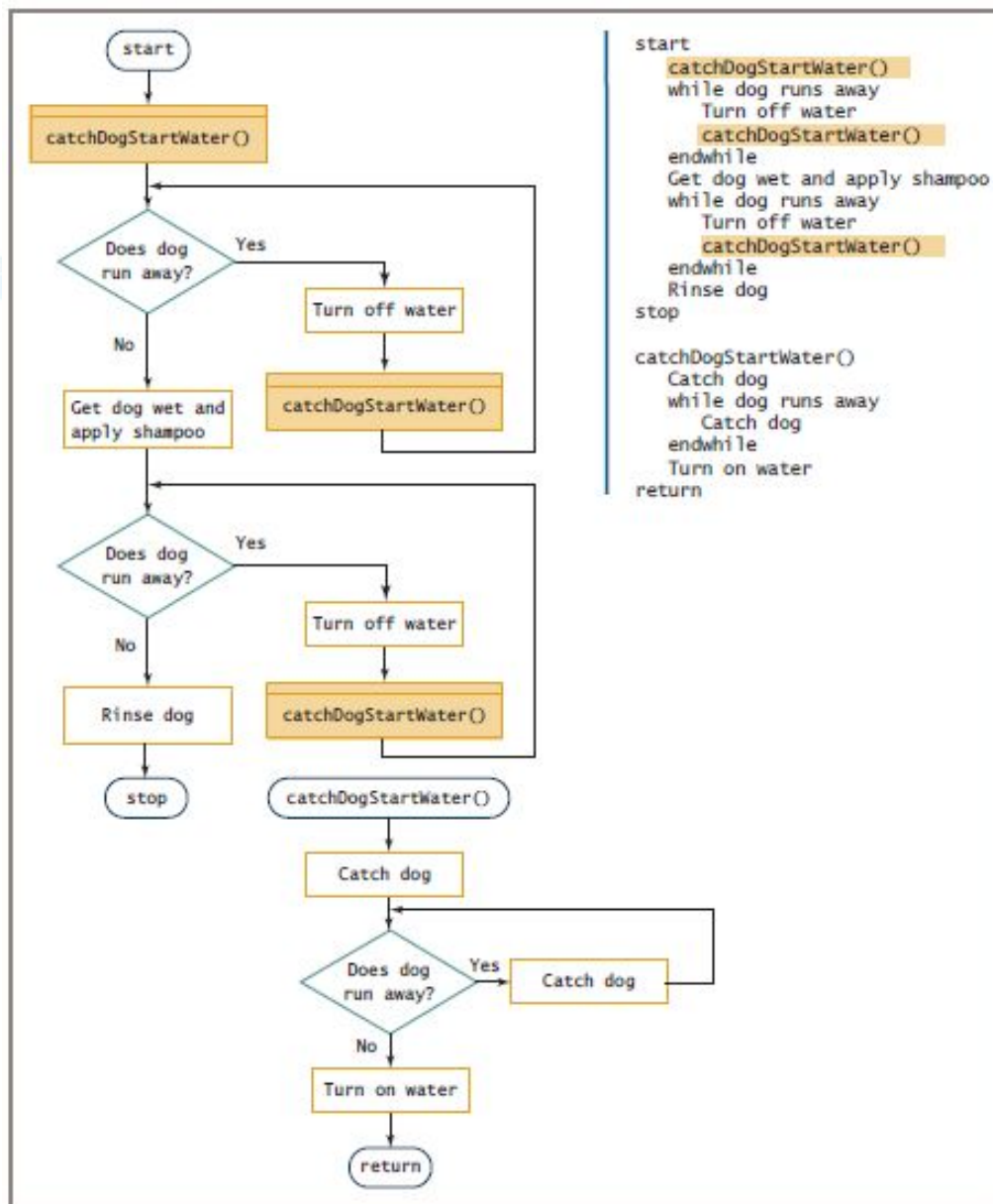


Figure 3-24 Modularized version of the dog-washing program



Summary

- Spaghetti code
 - Statements that do not follow rules of structured logic
- Three basic structures
 - Sequence, selection, and loop
 - Combined by stacking and nesting
- Priming input
 - Statement that reads the first input value prior to starting a structured loop



Summary (continued)

- Structured techniques promote:
 - Clarity
 - Professionalism
 - Efficiency
 - Modularity
- Flowcharts can be made structured by untangling
- Logical steps can be rewritten to conform to the three structures