Chapter 1

I Am *Not* a Control Freak! (Control Structures)

At a Glance

Instructor's Manual Table of Contents

Lecture Notes

Overview

This chapter introduces the three control structures used in programming. A park bench is used to illustrate sequence and selection and repetition.

Chapter Objectives

After studying Chapter 1, students should be able to:

- Describe the three control structures
- Write simple algorithms using the sequence, selection, and repetition structures

Teaching Tips

Control Structures

1. Explain that all computer programs, no matter how simple or how complex, are written using one or more of three basic structures: sequence, selection, and repetition. These structures are called **control structures** or **logic structures** because they control the flow of a program's logic.

The Sequence Structure

1. Introduce the **sequence structure** in a computer program. This structure directs the computer to process the program instructions, one after another, in the order listed in the program. Note that the sequence structure can be found in every program.

2. Explain that **algorithms** are a set of step-by-step instructions that accomplish a task.

Teaching Tip	Have the students write some algorithms to accomplish simple tasks. Ask them, for example, to write the steps needed to make a sandwich….or to sign on to their computer. Use Figure 1-1 as an example of how to sit on a bench.

The Selection Structure

1. Describe the **selection structure** as a structure that makes a decision and then takes appropriate action based on that decision. Note that this is also called the **decision structure**.

2. Note the changes needed in the algorithm in Figure 2-2 that make it a decision.

The Repetition Structure

1. Explain that, when used in a program, the **repetition structure** directs the computer to repeat one or more instructions until some condition is met, at which time the computer should stop repeating the instructions. Note that the repetition structure is also referred to as a **loop** or **iteration**.

Teaching Tip	To learn more about control structures, refer students to the following Web site: http://msdn.microsoft.com/en-us/library/ca8tdhcs.aspx

Quick Quiz 1

1. A(n) ____ is a set of step-by-step instructions that accomplish a task.
   Answer: algorithm

2. The repetition structure is also referred to as a(n) ____.
   Answer: loop or iteration

3. In most programs, you will use both the ____ and ____ structures.
   Answer: selection, repetition

4. A cookie recipe provides a good example of the ____ structure.
   Answer: sequence

Class Discussion Topics

1. What is the purpose of a repetition structure?

2. Why are algorithms necessary?

Additional Projects

1. Ask your students to describe various means to transfer control out of a control structure.

2. Ask your students to write an algorithm that incorporates all three control structures.

Additional Resources

1. Decision structures:
   http://msdn.microsoft.com/en-us/library/hh892482.aspx

2. Loop structures:
   http://msdn.microsoft.com/en-us/library/ezk76t25.aspx

3. Algorithms:
   www.howstuffworks.com/question717.htm

Key Terms

- **Algorithm**—the set of step-by-step instructions that accomplish a task
- **Control structures**—the sequence, selection, and repetition structures, which control the flow of a program's logic; also called logic structures
- **Decision structure**—another term for the selection structure
- **Iteration**—another term for the repetition structure
- **Logic structures**—another term for control structures
- **Loop**—another term for the repetition structure
- **Repetition structure**—the control structure that directs the computer to repeat one or more instructions until some condition is met, at which time the computer should stop repeating the instructions; also called a loop or iteration
- **Selection structure**—the control structure that directs the computer to make a decision and then take the appropriate action based on that decision; also called the decision structure
- **Sequence structure**—the control structure that directs the computer to process each instruction in the order listed in the program