true, the program executes the statement(s) in the body. As with a for loop, the body consists of a single statement or a block defined by paired braces. After it finishes with the body, the program returns to the test-condition and reevaluates it. If the condition is nonzero, the program executes the body again. This cycle of testing and execution continues until the test-condition evaluates to false. (See Figure 5.3.) Clearly, if you want the loop to terminate eventually, something within the loop body must do something to affect the test-condition expression. For example, the loop can increment a variable used in the test condition or read a new value from keyboard input. Like the for loop, the while loop is an entry-condition loop. Thus, if the test-condition evaluates to false at the beginning, the program never executes the body of the loop.

Figure 5.3. The while loop.