The Logical NOT Operator: !

The ! operator negates, or reverses the truth value of, the expression that follows it. That is, if expression is true, then !expression is false, and vice versa. More precisely, if expression is true or nonzero, then !expression is false. Incidentally, many people call the exclamation point bang, making !x bang-exe and !!x bang-bang-exe.

Usually you more clearly can express a relationship without using this operator:

```
if (!(x > 5)) // if (x <= 5) is clearer
```

But the ! operator can be useful with functions that return true-false values or values that can be interpreted that way. For example, `strcmp(s1,s2)` returns a nonzero (true) value if the two strings `s1` and `s2` are different from each other and a zero value if they are the same. This implies that `!strcmp(s1,s2)` is true if the two strings are equal.

Listing 6.7 uses this technique (applying the ! operator to a function return value) to screen numeric input for suitability to be assigned to type `int`. The user-defined function `is_int()`, which we discuss further in a moment, returns true if its argument is within the range of values assignable to type `int`. The program then uses the test `while(!is_int(num))` to reject values that don't fit in the range.

Listing 6.7 not.cpp

```
// not.cpp -- using the not operator
#include <iostream>
#include <climits>
using namespace std;
bool is_int(double);
int main()
{

double num;

    cout << "Yo, dude! Enter an integer value: ";
    cin >> num;
    while (!is_int(num))    // continue while num is not int-able
```