cout << ", angle = " << dapos.angle * Rad_to_deg;
cout << " degrees\n";
}

**Compatibility Note**

Some implementations still use math.h instead of the newer cmath header file. Some compilers require explicit instructions to search the math library. For example, older versions of g++ uses this command line:

g++ structfun.C -lm

Here is a sample run:

Enter the x and y values: **30 40**
distance = 50, angle = 53.1301 degrees
Next two numbers (q to quit): **-100 100**
distance = 141.421, angle = 135 degrees
Next two numbers (q to quit): **q**

**Program Notes**

We've already discussed the two functions, so let's review how the program uses cin to control a while loop:

while (cin >> rplace.x >> rplace.y)

Recall that cin is an object of the istream class. The extraction operator (>>) is designed in such a way that cin >> rplace.x also is an object of that type. As you'll see in Chapter 11, "Working with Classes," class operators are implemented with functions. What really happens when you use cin >> rplace.x is that the program calls a function that returns a type istream value. Apply the extraction operator to the cin >> rplace.x object (as in cin >> rplace.x >> rplace.y), and you again get an object of the istream class. Thus, the entire while loop test expression eventually evaluates to cin, which, as you may recall, when used in the context of a test expression is converted to a bool value of true or false,