As with single arguments, you don't have to use the same variable names in the prototype as in the definition, and you can omit the variable names in the prototype:

void n_chars(char, int);  // prototype, style 2

However, providing variable names can make the prototype more understandable, particularly if two parameters are the same type. Then, the names can remind you which argument is which:

double melon_density(double weight, double volume);

**Listing 7.3** shows an example of a function with two arguments. It also illustrates how changing the value of a formal parameter in a function has no effect on the data in the calling program.

**Listing 7.3 twoarg.cpp**

```cpp
// twoarg.cpp -- a function with 2 arguments
#include <iostream>
using namespace std;
void n_chars(char, int);
int main()
{
    int times;
    char ch;

    cout << "Enter a character: ";
    cin >> ch;
    while (ch != 'q')  // q to quit
    {
        cout << "Enter an integer: ";
        cin >> times;
        n_chars(ch, times);  // function with two arguments
        cout << "\nEnter another character or press the" 
            " q-key to quit: ";
```