Only trailing parameters of a function can have default values. Some examples are:

```
void foo(int i, int j = 7);                    //legal
void goo(int i = 3, int j);                    //illegal
void hoo(int i, int j = 3, int k = 7);         //legal
void moo(int i = 1, int j = 2, int k = 3); //legal
void noo(int i, int j = 2, int k);             //illegal
```

3.6 Functions as Arguments

Functions in C++ can be thought of as the address of the compiled code residing in memory. They are therefore a form of pointer (see Section 3.11, "Pointer Types," on page 85), and can be passed as a pointer-value argument into another function. Using this idea, we write code that will print n values of a function starting at some initial value using a specific increment. This form of plotting function can be useful to generate a map of a function that will later be used to find properties of the function, such as its root.

In file root.cpp

```
#include <iostream.h>

double f(double x)
{
   return (x*x + 1.0/x);
}

void plot(double fcn(double), double x0, double incr, int n)
{
   for (int i = 0; i < n; ++i){
      cout << " x :" << x0
           << "   f(x) : " << fcn(x0) << endl;
      x0 += incr;
   }
}

int main()
{
   cout << "mapping function x*x + 1.0/x " << endl;
   plot(f, 0.01, 0.01, 100);
}
```