On machines with 2-byte ints, the last two formats must be changed to %lx and %ld, respectively. The functions printf() and scanf() use the conversion characters and d, x, and o in conversion specifications for decimal, hexadecimal, and octal, respectively. With printf(), formats of the form %x and %o cause integers to be printed out in hexadecimal and octal notation, but not prefaced with 0x or 0. The formats %#x and %#o can be used to get the prefixes. (See Section 11.1, "The Output Function printf()," on page 493, for further discussion.) *Caution:* When using scanf() to read in a hexadecimal number, do not type an 0x prefix.

C.13 Summary

1 The fundamental data types are char, short, int, long, unsigned versions of these, and three floating types. The type char is a 1-byte integral type mostly used for representing characters.

2 The type int is designed to be the "natural" or "working" integral type. The other integral types such as short, long, and unsigned are provided for more specialized situations.

3 Three floating types, float, double, and long double, are provided to represent real numbers. Typically, a float is stored in 4 bytes and a double in 8 bytes. The number of bytes used to store a long double varies from one compiler to another. However, as compilers get updated, the trend is to store a long double in 16 bytes. The type double, not float, is the "working" type.

4 Unlike integer arithmetic, floating arithmetic is not always exact. Engineers and numerical analysts often have to take roundoff effects into account when doing extensive calculations with floating-point numbers.

5 The unary operator sizeof can be used to find the number of bytes needed to store a type or the value of an expression. For example, sizeof(int) is 2 on some older small machines and is 4 on most new machines that have 32-bit words.

6 The usual mathematical functions, such as sin(), cos(), and tan(), are available in the mathematics library. Most of the functions in the library take a single argument of type double and return a value of type double. The standard header file *math.h* should be included when using these functions.

7 Automatic conversions occur in mixed expressions and across an equal sign. Casts can be used to force explicit conversions.

8 Integer constants beginning with 0x and 0 designate hexadecimal and octal integers, respectively.

9 Suffixes can be used to explicitly specify the type of a constant. For example, 3U is of type unsigned, and 7.0F is of type float.

10 A character constant such as 'A' is of type int in C, but it is of type char in C++. This is one of the few places where C++ differs from C.

Exercises

1 Not all real numbers are machine-representable; there are too many of them. Thus, the numbers that are available on a machine have a "graininess" to them. As an example of this, the code

```
double   x = 123.45123451234512345;
double   y = 123.45123451234512300;  /* last two digits zero */

printf("%.17f\n%.17f\n", x, y);
```

causes two identical numbers to be printed. How many zeros must the initializer for y end with to get different numbers printed? Explain your answer.

2 The mathematical formula

$$\sin^2(x) + \cos^2(x) = 1$$

holds for all x real. Does this formula hold on your machine? Try the following program:

```
#include <math.h>
#include <stdio.h>

int main(void)
{
  double   two_pi = 2.0 * M_PI;   /* in math.h */
  double   h      = 0.1;          /* step size */
  double   x;

  for (x = 0.0; x < two_pi; x += h)
    printf("%5.1f: %.15e\n",
       x, sin(x) * sin(x) + cos(x) * cos(x));
  return 0;
}
```

What happens if the format %.15e is changed to %.15f? Explain.