Okay. It isn't a "number type." It's a *numeric data type,* which is how you say "number type" if you work at the Pentagon. You have to tell the C compiler which type of number you're using because it thinks about numbers differently from the way humans do. For example, you have to know the following things about the number:

- Is it a whole number — without a fraction or decimal part?
- How big is the number (as in value-large, not big-on-the-page-large)?
- If the number does have a fractional part, how precise must the number be? (Like to the thousandths, millionths, or gazillionths decimal place. Scientists have to know the precision when they send missiles to countries with opposing ideologies.)

I know that this stuff is all alien to you. What most programmers want to do is say "I need a number variable — just give me one, quick — before this value slips out the back of the computer and becomes a government statistic!" But you have to think a little more before you do that.

- The most common numeric data type is the integer.
- If you're going to work with decimal numbers, such as a dollar amount, you need the *floating-point* number.
- Keep reading.

Numbers in C

A number of different types of numbers are used in C — different numeric data types, so to speak. Table 9-1 lists them all, along with other statistical information. Flag the table with a sticky note. This table is something you refer to now and again because only the truly insane would memorize it.

Table 9-1 C Numeric Data Types

Keyword	*Variable Type*	*Range*
char	Character (or string)	–128 to 127
int	Integer	–32,768 to 32,767
short	Short integer	–32,768 to 32,767
short int	Short integer	–32,768 to 32,767