

<i>Keyword</i>	<i>Variable Type</i>	<i>Range</i>
long	Long integer	-2,147,483,648 to 2,147,483,647
unsigned char	Unsigned character	0 to 255
unsigned int	Unsigned integer	0 to 65,535
unsigned short	Unsigned short integer	0 to 65,535
unsigned long	Unsigned long integer	0 to 4,294,967,295
float	Single-precision floating-point (accurate to 7 digits)	$\pm 3.4 \times 10^{38}$ to $\pm 3.4 \times 10^{-38}$
double	Double-precision floating-point (accurate to 15 digits)	$\pm 1.7 \times 10^{308}$ to $\pm 1.7 \times 10^{-308}$

- ✓ The *keyword* is the C language keyword used to declare the variable type. If you have been reading the chapters in this book in order, you have used `int`, `char`, and `float` already.
- ✓ The *variable type* tells you which type of variable the keyword defines. For example, `char` defines a character (or string) variable, and `int` defines integers. C has many variable types, each of which depends on the type of number or value being described.
- ✓ The *range* tells you how big of a number fits into the variable type. For example, integers range from -32,768 up to 0 and up again to 32,767. Other types of variables handle larger values. This value may be different on your compiler; use the values in Table 9-1 for reference only.

Table 9-1 isn't that complex. In all, C has really only four types of variables:

- ✓ `char`
- ✓ `int`
- ✓ `float`
- ✓ `double`

The `int` can be modified with either `short` or `long`, and both `char` and `int` are modified with `unsigned`. The `float` and `double` variables are both floating-point, though the values held by `double` are larger.