- ***Hexadecimal*** constants contain digits between `0` and `9` and letters between `a` and `f`, and always begin with `0x`:

  ```
  0xf   0xff   0x7fff
  ```

  The letters in a hexadecimal constant may be either upper or lower case:

  ```
  0xff   0xfF   0xFf   0xFF   0Xff   0XfF   0XFf   0XFF
  ```

Keep in mind that octal and hexadecimal are nothing more than an alternative way of writing numbers; they have no effect on how the numbers are actually stored. (Integers are always stored in binary, regardless of what notation we've used to express them.) We can switch from one notation to another at any time, and even mix them: `10 + 015 + 0x20` has the value 55 (decimal). Octal and hex are most convenient for writing low-level programs; we won't use these notations much until Chapter 20.

The type of a *decimal* integer constant is normally `int`. However, if the value of the constant is too large to store as an `int`, the constant has type `long int` instead. In the unlikely case that the constant is too large to store as a `long int`, the compiler will try `unsigned long int` as a last resort. The rules for determining the type of an *octal* or *hexadecimal* constant are slightly different: the compiler will go through the types `int`, `unsigned int`, `long int`, and `unsigned long int` until it finds one capable of representing the constant.

To force the compiler to treat a constant as a long integer, just follow it with the letter `L` (or `l`):

```
15L   0377L   0x7fffL
```

To indicate that a constant is unsigned, put the letter `U` (or `u`) after it:

```
15U   0377U   0x7fffU
```

`L` and `U` may be used in combination to show that a constant is both long *and* unsigned: `0xffffffffUL`. (The order of the `L` and `U` doesn't matter, nor does their case.)

C99

Integer Constants in C99

In C99, integer constants that end with either `LL` or `ll` (the case of the two letters must match) have type `long long int`. Adding the letter `U` (or `u`) before or after the `LL` or `ll` denotes a constant of type `unsigned long long int`.

C99's general rules for determining the type of an integer constant are a bit different from those in C89. The type of a decimal constant with no suffix (`U`, `u`, `L`, `l`, `LL`, or `ll`) is the "smallest" of the types `int`, `long int`, or `long long int` that can represent the value of that constant. For an octal or hexadecimal constant, however, the list of possible types is `int`, `unsigned int`, `long int`, `unsigned long int`, `long long int`, and `unsigned long long int`, in that order. Any suffix at the end of a constant changes the list of possible types. For