

 **练习题 2.44** 假设我们在对有符号值使用补码运算的 32 位机器上运行代码。对于有符号值使用的是算术右移，而对于无符号值使用的是逻辑右移。变量的声明和初始化如下：

```
int x = foo(); /* Arbitrary value */
int y = bar(); /* Arbitrary value */

unsigned ux = x;
unsigned uy = y;
```

对于下面每个 C 表达式，1) 证明对于所有的  $x$  和  $y$  值，它都为真(等于 1)；或者 2) 给出使得它为假(等于 0)的  $x$  和  $y$  的值：

- A.  $(x > 0) \parallel (x - 1 < 0)$
- B.  $(x \& 7) != 7 \parallel (x \ll 29 < 0)$
- C.  $(x * x) \geq 0$
- D.  $x < 0 \parallel -x \leq 0$
- E.  $x > 0 \parallel -x \geq 0$
- F.  $x + y == uy + ux$
- G.  $x \sim y + uy + ux == -x$

## 2.4 浮点数

浮点表示对形如  $V = x \times 2^y$  的有理数进行编码。它对执行涉及非常大的数字( $|V| \gg 0$ )、非常接近于 0( $|V| \ll 1$ )的数字，以及更普遍地作为实数运算的近似值的计算，是很有用的。

直到 20 世纪 80 年代，每个计算机制造商都设计了自己的表示浮点数的规则，以及对浮点数执行运算的细节。另外，它们常常不会太多地关注运算的精确性，而把实现的速度和简便性看得比数字精确性更重要。

大约在 1985 年，这些情况随着 IEEE 标准 754 的推出而改变了，这是一个仔细制订的表示浮点数及其运算的标准。这项工作是从 1976 年开始由 Intel 赞助的，与 8087 的设计同时进行，8087 是一种为 8086 处理器提供浮点支持的芯片。他们请 William Kahan(加州大学伯克利分校的一位教授)作为顾问，帮助设计未来处理器浮点标准。他们支持 Kahan 加入一个 IEEE 资助的制订工业标准的委员会。这个委员会最终采纳的标准非常接近于 Kahan 为 Intel 设计的标准。目前，实际上所有的计算机都支持这个后来被称为 IEEE 浮点的标准。这大大提高了科学应用程序在不同机器上的可移植性。

### 旁注 IEEE(电气和电子工程师协会)

电气和电子工程师协会(IEEE，读做“eye-triple-ee”)是一个包括所有电子和计算机技术的专业团体。它出版刊物，举办会议，并且建立委员会来定义标准，内容涉及从电力传输到软件工程。另一个 IEEE 标准的例子是无线网络的 802.11 标准。

在本节中，我们将看到 IEEE 浮点格式中数字是如何表示的。我们还将探讨舍入(rounding)的问题，即当一个数字不能被准确地表示为这种格式时，就必须向上调整或者向下调整。然后，我们将探讨加法、乘法和关系运算符的数学属性。许多程序员认为浮点数没意思，往坏了说，深奥难懂。我们将看到，因为 IEEE 格式是定义在一组小而一致的原则上的，所以它实际上是相当优雅和容易理解的。