

C语言中的 if-else 语句的通用形式模板如下：

```
if (test-expr)
    then-statement
else
    else-statement
```

这里 *test-expr* 是一个整数表达式，它的取值为 0(解释为“假”)或者为非 0(解释为“真”)。两个分支语句中(*then-statement* 或 *else-statement*)只会执行一个。

对于这种通用形式，汇编实现通常会使用下面这种形式，这里，我们用 C 语法来描述控制流：

```
t = test-expr;
if (!t)
    goto false;
then-statement
goto done;
false:
    else-statement
done:
```

也就是，汇编器为 *then-statement* 和 *else-statement* 产生各自的代码块。它会插入条件和无条件分支，以保证能执行正确的代码块。

旁注 用 C 代码描述机器代码

图 3-16 给出了一个示例，用来展示把 C 语言控制结构翻译成机器代码。图中包括示例的 C 函数 a 和由 GCC 生成的汇编代码的注释版本 c，还有一个与汇编代码结构高度一致的 C 语言版本 b。机器代码的 C 语言表示有助于你理解其中的关键点，能引导你理解实际的汇编代码。

练习题 3.16 已知下列 C 代码：

```
void cond(long a, long *p)
{
    if (p && a > *p)
        *p = a;
}
```

GCC 会产生下面的汇编代码：

```
void cond(long a, long *p)
a in %rdi, p in %rsi
cond:
    testq    %rsi, %rsi
    je       .L1
    cmpq     %rdi, (%rsi)
    jge      .L1
    movq     %rdi, (%rsi)
.L1:
    rep; ret
```

- 按照图 3-16b 中所示的风格，用 C 语言写一个 goto 版本，执行同样的计算，并模拟汇编代码的控制流。像示例中那样给汇编代码加上注解可能会有所帮助。
- 请说明为什么 C 语言代码中只有一个 if 语句，而汇编代码包含两个条件分支。