Q: I wanted to compile a program that I hadn't finished writing, so I "conditioned out" the unfinished part:

```
#if 0
...
#endif
```

When I compiled the program, I got an error message referring to one of the lines between `#if` and `#endif`. Doesn't the preprocessor just ignore these lines? [p. 338]

A: No, the lines aren't completely ignored. Comments are processed before preprocessing directives are executed, and the source code is divided into preprocessing tokens. Thus, an unterminated comment between `#if` and `#endif` may cause an error message. Also, an unpaired single quote or double quote character may cause undefined behavior.

Exercises

Section 14.3

1. Write parameterized macros that compute the following values.
   (a) The cube of `x`.
   (b) The remainder when `n` is divided by 4.
   (c) 1 if the product of `x` and `y` is less than 100, 0 otherwise.

   Do your macros always work? If not, describe what arguments would make them fail.

W 2. Write a macro `NELEMS(a)` that computes the number of elements in a one-dimensional array `a`. *Hint:* See the discussion of the `sizeof` operator in Section 8.1.

3. Let `DOUBLE` be the following macro:

   ```
   #define DOUBLE(x) 2*x
   ```

   (a) What is the value of `DOUBLE(1+2)`?
   (b) What is the value of `4/DOUBLE(2)`?
   (c) Fix the definition of `DOUBLE`.

W 4. For each of the following macros, give an example that illustrates a problem with the macro and show how to fix it.

   (a) `#define AVG(x,y)  (x+y)/2`

   (b) `#define AREA(x,y)  (x)*(y)`

W *5. Let `TOUPPER` be the following macro:

   ```
   #define TOUPPER(c) ('a'<=(c)&&(c)<='z'?(c)-'a'+'A':(c))
   ```

   Let `s` be a string and let `i` be an `int` variable. Show the output produced by each of the following program fragments.

   (a)
   ```
   strcpy(s, "abcd");
   i = 0;
   putchar(TOUPPER(s[++i]));
   ```