Indeed the code is explained, probably for pedagogical reasons, but a bad practice well commented remains bad. (The last lines of the original maze program in Chapter 4 contain a similar instance.)

Another example where potentially dangerous code is treated with a comment instead of a rewrite is this fragment:

```
      6 E=E+.5
C       TEST FOR VOLTAGE EXCEEDING 3.0.
        IF(E-3.01)5,7,7
```

Since the comment and the code disagree, something is afoot that we are not being told about. Why is the test against 3.01 instead of 3.0? The most likely explanation is that it defends against some form of floating point rounding error, but in the absence of a useful comment, we can only guess. The way to treat this situation is not by adding arbitrary unexplained tolerances to tests. If the code has to be this way, explain it, for it is certainly not obvious. But if (as seems more likely) it reflects a poor algorithm, *change the code.*

Don't comment bad code — rewrite it.

Variable names, labels, and even Fortran statement numbers can aid or hinder documentation. Well-chosen names jog the memory; too-similar or meaningless identifiers hamper understanding. For example:

```
      LOGICAL EL,EM,EN,AKK,ELL,EMM,ENN,ELLL,EMMM,ENNN,ELLLL,EMMMM
      ...
      EL = A.EQ.5.*C
      EM = B.EQ.A+C
      EN = C/B.EQ.C/A
      AKK = A/B.EQ.B/C
      ELL = A/B.EQ.C
      EMM = B.LT.A
      ENN = C.GT.B
      ELLL = A.GT.C
      EMMM = A.GE.B*C
      ENNN = EM.OR.EN
      ELLLL = EN.AND.AKK
      EMMMM = .NOT.(EL.AND.EN)
      PRINT 20,A,B,C,EL,EM,EN,AKK,ELL,EMM,ENN,ELLL,EMMM,ENNN,ELLLL,EMMMM
   20 FORMAT (1X,3F10.1,12L5)
```

These names have no mnemonic significance — even AKK, although different from the others for no apparent reason, conveys no information. The similarities invite misunderstanding and typing errors.

One solution might be names reminiscent of the test performed, such as

```
BLTA = B .LT. A
```

but since this seems strained, it is probably easiest to make an array called E and put headings on the output. Then the FORMAT statement can serve as part of the documentation.