Choose a data representation
that makes the program simple.

Excessive use of labels (statement numbers) and GOTO's is often the hallmark of undisciplined design, a sign that a program is out of control. Tracing the flow can be next to impossible if there are too many potential paths from one point to another. Even when such code is correct, it is hard to understand and thus even harder to modify. Consider this program for converting a year and day of the year into the month and day of the month:

```
DATES:  PROC OPTIONS (MAIN);
READ:   GET DATA (IYEAR, IDATE);
        IF IDATE < 1 | IDATE > 366 | IYEAR < 0 THEN RETURN;
        IF IDATE <= 31 THEN GO TO JAN;
        L = 1;
        I = IYEAR/400; IF I = IYEAR/400 THEN GO TO LEAP;
        I = IYEAR/100; IF I = IYEAR/100 THEN GO TO NOLEAP;
        I = IYEAR/4; IF I = IYEAR/4 THEN GO TO LEAP;
NOLEAP: L = 0;
        IF IDATE > 365 THEN RETURN;
 LEAP:  IF IDATE > 181 + L THEN GO TO G181;
        IF IDATE > 90 + L THEN GO TO G90;
        IF IDATE > 59 + L THEN GO TO G59;
        MONTH = 2; IDAY = IDATE - 31; GO TO OUT;
  G59:  MONTH = 3; IDAY = IDATE - (59 + L); GO TO OUT;
  G90:  IF IDATE > 120 + L THEN GO TO G120;
        MONTH = 4; IDAY = IDATE - (90 + L); GO TO OUT;
 G120:  IF IDATE > 151 + L THEN GO TO G151;
        MONTH = 5; IDAY = IDATE - (120 + L); GO TO OUT;
 G151:  MONTH = 6; IDAY = IDATE - (151 + L); GO TO OUT;
 G181:  IF IDATE > 273 + L THEN GO TO G273;
        IF IDATE > 243 + L THEN GO TO G243;
        IF IDATE > 212 + L THEN GO TO G212;
        MONTH = 7; IDAY = IDATE - (181 + L); GO TO OUT;
 G212:  MONTH = 8; IDAY = IDATE - (212 + L); GO TO OUT;
 G243:  MONTH = 9; IDAY = IDATE - (243 + L); GO TO OUT;
 G273:  IF IDATE > 334 + L THEN GO TO G334;
        IF IDATE > 304 + L THEN GO TO G304;
        MONTH = 10; IDAY = IDATE - (273 + L); GO TO OUT;
 G304:  MONTH = 11; IDAY = IDATE - (304 + L); GO TO OUT;
 G334:  MONTH = 12; IDAY = IDATE - (334 + L);
OUT:    PUT DATA (MONTH,IDAY,IYEAR) SKIP;
        GO TO READ;
JAN:    MONTH=1; IDAY=IDATE; GO TO OUT;
        END DATES;
```

We have nothing to say about the "structure" of this program; the code speaks for itself. There is one curious usage that bears explaining, however, in the leap year determination. Three lines have the form

```
I = IYEAR/n; IF I = IYEAR/n THEN GO TO label;
```

It would seem that the GOTO is always obeyed, since I surely equals that which was