

It is easy to overlook a poor choice of data representation by getting involved in fixing up the intricate code that usually accompanies it. The following program reads cards and centers the non-blank information on each card within a border of periods.

```

        DIMENSION STRING(80),PHRASE(80)
        BLANK = 1H
        PERIOD = 1H.
    999 READ 100, (PHRASE(I),I=1,80)
    100 FORMAT(80A1)
    103 FORMAT(1H0)
C
C      LOOP 1
C
        NBEF=0
        J =1
    1  IF(PHRASE(J).NE.BLANK) GO TO 2
        NBEF=NBEF+1
        J = J+1
        IF(J.EQ.81) GO TO 1000
        GO TO 1
C
C      LOOP 2
C
    2  NAFT=0
        I=80
    3  IF(PHRASE(I).NE.BLANK)GO TO 4
        NAFT=NAFT+1
        I=I-1
        GO TO 3
C
C      COMPUTE LENGTHS OF PHRASE AND MARGIN
C
    4  LENGTH=80-(NBEF+NAFT)
        MARGIN = (80-LENGTH)/2+1
        IND1=NBEF
        IND2=MARGIN
        DO 41 J=2,79
    41  STRING(J)=BLANK
C
C      TRANSFER PHRASE TO STRING
C
        DO 5 I=1,LENGTH
            IND1=IND1+1
            IND2=IND2+1
    5  STRING(IND2) = PHRASE(IND1)
        STRING(1)=PERIOD
        STRING(80) = PERIOD
        PRINT 103
        PRINT 101,(STRING(I),I=1,80)
    101 FORMAT(1H , 80(1H.)/3(2H .,78X,1H./),1H ,80A1/3(2H .,78X,1H./)
        1,1H ,80(1H.))
        GO TO 999
    1000 PRINT 102
    102 FORMAT(1X, 37HBLANK CARD READ IN. EXIT FROM CENTER.)
        CALL EXIT
        END

```

Let us first examine the two errors that mar this otherwise straightforward program. The loop that ends at statement 5 copies the non-blank part of `PHRASE` into the appropriate part of `STRING`. Then positions 1 and 80 of the output area, `STRING`, are overwritten with a `PERIOD`, regardless of the length of the input. This bodes ill for input strings 79 or 80 characters long. We can avoid the overwrite by