Normal Family Distribution Functions: FORTRAN and BASIC Programs

R. J. CRAIG

A.T.&T. Bell Laboratories, Naperville, Illinois 60566

This paper provides a set of FORTRAN and BASIC subroutines that calculate the cumulative probability distribution functions for the standard normal, chi-square, *F*, and Student's *t* distributions.

Editor's Introduction

Many programs submitted to *JQT* for publication require the calculation of probabilities and percentage points for standard probability distributions. The quality of the submitted algorithms has been highly variable. Further, a literature search has revealed that many published algorithms contain subtle inaccuracies. Thus, it seemed wise to provide a series of fast and accurate algorithms for our readers that can be used as standards for future *JQT* computer program articles.

This paper is the first in a series of articles that will provide FORTRAN and BASIC subroutines for the computation of probabilities and percentage points of selected distributions. It provides a set of FORTRAN and BASIC subroutines that calculate the cumulative probability distribution functions (cdf's) for the normal distribution and the distributions related to it, namely, the chi-square, *F*, and Student's *t*. The programs do not strive for extreme accuracy, but rather compromise between the needs for reasonable accuracy and speed of execution. Thus the programs are guaranteed to provide 16 decimal places of accuracy, i.e., 16 positions after the decimal point.

Dr. Craig is a Member of the Technical Staff at the Indian Hill South A.T.&T. Bell Laboratories site.

KEY WORDS: Chi-square Distribution, *F* Distribution, Normal Distribution, Student's *t* Distribution

Program Descriptions

The subroutines for the standard normal cdf (FORTRAN Listing 1; BASIC Listing 1) use the relationship between the standard normal distribution and the error function

$$\Phi(x) = \begin{cases} \frac{1}{2}[1 + \mathrm{erf}(x/\sqrt{2})] & \text{if} \quad x \geq 0 \\ \frac{1}{2}[1 - \mathrm{erf}(x/\sqrt{2})] & \text{if} \quad x < 0 \end{cases}$$

and an approximation due to Strecok (1968) for the error function

$$\mathrm{erf}(x) \approx \frac{2}{\pi}\left[\frac{x}{5} + \sum_{n=1}^{37} \exp\{-(n/5)^2\}\sin(2nx/5)/n\right]$$

$$\text{for } |x| \leq \frac{5\pi}{2}.$$

The subroutines for the chi-square cdf with integer degrees of freedom (FORTRAN Listing 2; BASIC Listing 2) are based on an algorithm due to Hill and Pike (1967). Denoting by χ_n^2 a chi-square random variable with n degrees of freedom, they use the recurrence formula

$$\Pr[\chi_n^2 > x] = \Pr[\chi_{n-2}^2 > x] + \frac{(x/2)^{(n-2)/2}e^{-x/2}}{\Gamma(n/2)}$$

where

$$\Pr[\chi_2^2 > x] = e^{-x/2}$$
$$\Pr[\chi_1^2 > x] = 2\Phi(-\sqrt{x}).$$

The subroutines for the F cdf with integer degrees of freedom (FORTRAN Listing 3; BASIC Listing 3) are based on an algorithm due to Dorrer (1968). Denoting by $F_{m,n}$ an F random variable with numerator degrees of freedom m and denominator degrees of freedom n, they use the recurrence formulas

$$\begin{aligned}\Pr[F_{m,n} \le x] &= \Pr\left[F_{m,n-2} \le \left(\frac{n-2}{n}\right) x\right] \\ &\quad + g(n, m, z) \\ &= \Pr\left[F_{m-2,n} \le \left(\frac{m}{m-2}\right) x\right] \\ &\quad - g(m, n, w)\end{aligned}$$

where

$$w = x(m/n)$$
$$z = (n/m)/x$$

$$g(m, n, y) = \frac{2\Gamma((m+n)/2)\, y^{(m-2)/2}}{\Gamma(m/2)\Gamma(n/2)(y+1)^{(m+n-2)/2}(m+n-2)}$$

$$\Pr[F_{2,2} \le x] = \frac{w}{w+1}$$

$$\Pr[F_{2,1} \le x] = 1 - \frac{1}{\sqrt{w+1}}$$

$$\Pr[F_{1,2} \le x] = \sqrt{\frac{w}{w+1}}$$

$$\Pr[F_{1,1} \le x] = \frac{2}{\pi}\arctan(\sqrt{w}).$$

The subroutines for the Student's t cdf (FORTRAN Listing 4; BASIC Listing 4) are based on the relationship between Student's t and the F distribution. Denoting by t_n a t random variable with n degrees of freedom, the relationship is

$$\Pr[t_n \le t] = \begin{cases} \dfrac{1 + \Pr[F_{1,n} \le t^2]}{2} & \text{if } t \ge 0 \\[2ex] \dfrac{1 - \Pr[F_{1,n} \le t^2]}{2} & \text{if } t < 0. \end{cases}$$

Subroutines to deal with non-integer degrees of freedom will be provided in a forthcoming article.

Program Operation

The subroutines provided are meant to be inserted in (or linked with) programs written by the user. For BASIC programs care must be taken to not redefine elsewhere in the program the variables used in the subroutines. In addition some BASIC compiler/interpreters do not accept numeric values with more than eight digits. For compiler/interpreters of that kind, truncation of the values is the appropriate solution.

Examples

Example 1

The simple BASIC program given in Input Listing 1 can be used to generate values of the normal cdf. Output from the program is given in Output Listing 1.

INPUT LISTING 1. Listing for Example 1

```
00010 PRINT
00020 PRINT "INPUT X";
00030 INPUT X
00040 GOSUB 23000
00050 PRINT "THE PROBABILITY OF A STANDARD NORMAL"
00060 PRINT "RANDOM VARIABLE BEING <= X IS ";
00070 REM
00080 REM   THE PRINT USING STATEMENT IS NOT PART OF
00090 REM   THE ANSI STANDARD, BUT IT IS NECESSARY
00100 REM   ON SOME SYSTEMS TO OBTAIN MORE THAN SEVEN
00110 REM   SIGNIFICANT FIGURES.
00120 REM
00130 PRINT USING "#.################",P
00140 IF X<>0 THEN 10
00150 GO TO 23024
23000 REM
23001 REM   SUBROUTINE TO COMPUTE NORMAL CDF GOES HERE
23002 REM
```

OUTPUT LISTING 1. Listing for Example 1

```
INPUT X? 1
THE PROBABILITY OF A STANDARD NORMAL
RANDOM VARIABLE BEING <= X IS 0.8413447460685430

INPUT X? 2
THE PROBABILITY OF A STANDARD NORMAL
RANDOM VARIABLE BEING <= X IS 0.9772498680518208

INPUT X? 3
THE PROBABILITY OF A STANDARD NORMAL
RANDOM VARIABLE BEING <= X IS 0.9986501019683700

INPUT X? -.5
THE PROBABILITY OF A STANDARD NORMAL
RANDOM VARIABLE BEING <= X IS 0.3085375387259869

INPUT X? -5.8
THE PROBABILITY OF A STANDARD NORMAL
RANDOM VARIABLE BEING <= X IS 0.0000000033157459

INPUT X? 0
THE PROBABILITY OF A STANDARD NORMAL
RANDOM VARIABLE BEING <= X IS 0.5000000000000000
```

Example 2

Olsson (1973) used an approximation for the F cdf in his analysis of variance program, and in his numerical example the approximate probability of exceeding the F-ratio was computed as 0.146. If more accuracy is desired, the approximation routine could be replaced with the subroutine in FORTRAN Listing 3. The resulting probability of exceeding the F-ratio would then (with the appropriate format change) be the same as obtained by running the short program given in Input Listing 2, namely .147225014.

INPUT LISTING 2. Listing for Example 2

```
      DOUBLE PRECISION MST,MSE,Y,DFSHER
      DATA MST,MSE/1164.548096,528.474854/
      F=MST/MSE
      Y=DFSHER(F,2,14)
      Y=1.D0-Y
      WRITE(6,100) Y
100   FORMAT('0PROBABILITY OF EXCEEDING F-RATIO IS',F12.9)
      STOP
      END
C
C  INSERT THE F CDF (FUNCTION DFSHER) HERE
C  OR LINK THIS PROGRAM WITH THE FUNCTION.
C
```

Example 3

The multivariate T^2 statistic has application in the quality control field (see Kramer (1969, 1970) and Jackson (1980)), but tables of its distribution are not readily available. The FORTRAN program in Input Listing 3 computes the cdf of T^2 using the relationship with the F distribution

$$\Pr[T^2_{p,n} \leq x] = \Pr\left[F_{p,n-p} \leq \left\{\frac{n-p}{P(n-1)}\right\}x\right]$$

where

p = number of variables in T^2

n = number of observations on each variable.

The results from running the program are shown in Output Listing 2.

INPUT LISTING 3. Listing for Example 3

```
      DOUBLE PRECISION TSQRD,X
      INTEGER P
      DATA P,N/5,26/
      WRITE(6,100)
      DO 1 I=1,20
      X=I
      Y=TSQRD(X,P,N)
      WRITE(6,101) P,N,X,Y
1     CONTINUE
100   FORMAT('0TSQUARED CDF'//' P    N      X',
     &'    PR[TSQUARED(P,N)<=X]')
101   FORMAT(1H ,I1,3X,I2,3X,F4.1,2X,F18.16)
      STOP
      END
      DOUBLE PRECISION FUNCTION TSQRD(X,P,N)
      DOUBLE PRECISION X,Y,DFSHER
      INTEGER P
      M=N-P
      Y=M*X/(P*(N-1))
      TSQRD=DFSHER(Y,P,M)
      RETURN
      END
C
C  INSERT THE F CDF (FUNCTION DFSHER) HERE
C  OR LINK THIS PROGRAM WITH THE FUNCTION.
C
```

OUTPUT LISTING 2. Listing for Example 3

```
TSQUARED CDF

P    N      X    PR[TSQUARED(P,N)<=X]
5    26    1.0   0.0284001547843218
5    26    2.0   0.1146319881081581
5    26    3.0   0.2301270067691803
5    26    4.0   0.3508886098861694
5    26    5.0   0.4636077582836151
5    26    6.0   0.5624989271163940
5    26    7.0   0.6461088061332703
5    26    8.0   0.7151811122894287
5    26    9.0   0.7714052200317383
5    26   10.0   0.8167402744293213
5    26   11.0   0.8530800687210083
5    26   12.0   0.8821121454238892
5    26   13.0   0.9052655696868896
5    26   14.0   0.9237235188484192
5    26   15.0   0.9384455084800720
5    26   16.0   0.9502013921737671
5    26   17.0   0.9596043825149536
5    26   18.0   0.9671406149864197
5    26   19.0   0.9731945395469666
5    26   20.0   0.9780697226524353
```

References

DORRER, E. (1968). "Algorithm 322: F-distribution". *Communications of the Association for Computing Machinery* 11, p. 116.

HILL, I. D. and PIKE, C. (1967). "Algorithm 299: Chi-squared Integral". *Communications of the Association for Computing Machinery* 10, p. 243.

KRAMER, C. Y. and JENSEN, D. R. (1969, 1970). "Fundamentals of Multivariate Analysis, Parts I–IV". *Journal of Quality Technology* 1, pp. 120–133, 264–277; Vol. 2, pp. 32–40.

JACKSON, J. E. (1980). "Principal Components and Factor Analysis: Part I—Principal Components". *Journal of Quality Technology* 12, pp. 201–213.

OLSSON, D. M. (1973). "One-Way Analysis of Variance". *Journal of Quality Technology* 5, pp. 191–193.

STRECOK, A. J. (1968). "On Calculation of the Inverse of the Error Function". *Mathematics of Computation* 22, pp. 144–158.

FORTRAN Listing 1.

```
      DOUBLE PRECISION FUNCTION DNML(X)
C
C  COMPUTES THE CUMULATIVE DISTRIBUTION FUNCTION
C  P[Y<=X] OF A RANDOM VARIABLE Y HAVING A
C  STANDARD NORMAL DISTRIBUTION.
C
      DOUBLE PRECISION X,Y,S,RN,ZERO,ONE,ERF,SQRT2,PI
      DATA SQRT2,ONE/1.414213562373095,1.D0/
      DATA PI,ZERO/3.141592653589793,0.D0/
      Y=X/SQRT2
      IF(X.LT.ZERO) Y=-Y
      S=ZERO
      DO 1 N=1,37
      RN=DFLOAT(N)
      S=S+DEXP(-RN*RN/25)/N*DSIN(2*N*Y/5)
1     CONTINUE
      S=S+Y/5
      ERF=2*S/PI
      DNML=(ONE+ERF)/2
      IF(X.LT.ZERO) DNML=(ONE-ERF)/2
      IF(X.LT.-8.3D0) DNML=ZERO
      IF(X.GT.8.3D0) DNML=ONE
      RETURN
      END
```

FORTRAN Listing 2.

```
      DOUBLE PRECISION FUNCTION DCHISQ(X,N)
C
C  COMPUTES THE CUMULATIVE DISTRIBUTION FUNCTION
C  P[Y<=X] OF A RANDOM VARIABLE Y HAVING A CHI
C  SQUARED DISTRIBUTION WITH N DEGREES OF FREEDOM
C  USING THE FUNCTION DNML.
```

```
C
      DOUBLE PRECISION A,Y,S,E,C,Z,X1,X,DNML
      DOUBLE PRECISION PI,ONE,ZERO,HALF,LARGEX
      LOGICAL BIGX
C
C  LARGEX IS THE LARGEST VALUE OF X SUCH THAT DEXP(-X/2)
C  IS ACCURATE FOR YOUR PARTICULAR MACHINE.
C
      DATA LARGEX,PI/174.99646DØ,3.141592653589793/
      DATA ONE,ZERO,HALF/1.DØ,Ø.DØ,.5DØ/
      IF(X.LT.Ø.DØ) WRITE(6,1ØØ) 'X',Ø
      IF(N.LT.1) WRITE(6.1ØØ) 'N',1
      IF(X.GE.Ø.DØ.AND.N.GE.1) GOTO 1Ø
      DCHISQ=-ONE
      RETURN
1Ø    A=HALF*X
      I=MOD(N+1,2)
      BIGX=.FALSE.
      IF(X.GT.LARGEX) BIGX=.TRUE.
      Y=DEXP(-A)
      IF(N.EQ.1.OR.BIGX) Y=ZERO
      S=Y
      IF(I.EQ.Ø) S=2*DNML(-DSQRT(X))
      IF(N.EQ.1) GOTO 3Ø
      X1=HALF*(N-1)
      Z=(ONE+I)/2
      IF(.NOT.BIGX) GOTO 4Ø
      E=ZERO
      IF(I.EQ.Ø) E=DLOG(DSQRT(PI))
      C=DLOG(A)
2Ø    IF(Z.GT.X1) GOTO 3Ø
      E=E+DLOG(Z)
      S=S+DEXP(C*Z-A-E)
      Z=Z+ONE
      GOTO 2Ø
3Ø    DCHISQ=ONE-S
      RETURN
4Ø    E=ONE
      IF(I.EQ.Ø) E=ONE/DSQRT(PI*A)
      C=ZERO
5Ø    IF(Z.GT.X1) GOTO 6Ø
      E=E*A/Z
      C=C+E
      Z=Z+ONE
      GOTO 5Ø
6Ø    DCHISQ=ONE-C*Y-S
      RETURN
1ØØ   FORMAT(' IN THE FUNCTION DCHISQ(X,N) THE ',
     &'PARAMETER ',A1,' MUST BE >= ',I1)
      END
```

FORTRAN Listing 3.

```
      DOUBLE PRECISION FUNCTION DFSHER(F,M,N)
C
C  COMPUTES THE CUMULATIVE DISTRIBUTION FUNCTION
C  P[Y<=F] OF A RANDOM VARIABLE Y HAVING AN F
C  DISTRIBUTION WITH NUMERATOR DEGREES OF FREEDOM
C  M AND DENOMINATOR DEGREES OF FREEDOM N.
C
      DOUBLE PRECISION F,W,Y,Z,D,P,ZK,PI,ONE,ZERO
      DATA PI,ONE,ZERO/3.141592653589793,1.DØ,Ø.DØ/
      IF(F.LT.Ø.DØ) WRITE(6,1ØØ) 'F',Ø
      IF(M.LT.1) WRITE(6,1ØØ) 'M',1
      IF(N.LT.1) WRITE(6,1ØØ) 'N',1
      IF(F.GE.Ø.DØ.AND.M.GE.1.AND.N.GE.1) GOTO 5
      DFSHER=-ONE
      RETURN
5     K=2*(M/2)-M+2
      L=2*(N/2)-N+2
      W=F*M/N
      Z=ONE/(ONE+W)
      IF(K.NE.1) GOTO 2Ø
      IF(L.NE.1) GOTO 1Ø
      P=DSQRT(W)
      D=Z/P/PI
      P=2*DATAN(P)/PI
      GOTO 4Ø
1Ø    P=DSQRT(W*Z)
      D=P*Z/W/2
      GOTO 4Ø
2Ø    IF(L.NE.1) GOTO 3Ø
      P=DSQRT(Z)
      D=Z*P/2
      P=ONE-P
      GOTO 4Ø
3Ø    D=Z*Z
      P=W*Z
4Ø    Y=2*W/Z
      IF(K.NE.1) GOTO 6Ø
      J=L+2
5Ø    IF(J.GT.N) GOTO 7Ø
      D=(ONE+DFLOAT(K)/(J-2))*D*Z
      P=P+D*Y/(J-1)
      J=J+2
      GOTO 5Ø
6Ø    ZK=Z**((N-1)/2)
      D=D*ZK*N/L
      P=P*ZK+W*Z*(ZK-ONE)/(Z-ONE)
7Ø    Y=W*Z
      Z=2/Z
      L=N-2
      I=K+2
8Ø    IF(I.GT.M) GOTO 9Ø
      J=I+L
      D=Y*D*J/(I-2)
      P=P-Z*D/J
      I=I+2
      GOTO 8Ø
9Ø    DFSHER=P
      IF(P.GT.ONE) DFSHER=ONE
      IF(P.LT.ZERO) DFSHER=ZERO
      RETURN
1ØØ   FORMAT(' IN THE FUNCTION DFSHER(F,M,N) THE ',
     &'PARAMETER ',A1,' MUST BE >= ',I1)
      END
```

FORTRAN Listing 4.

```
      DOUBLE PRECISION FUNCTION DSTDNT(T,N)
C
C  COMPUTES THE CUMULATIVE DISTRIBUTION FUNCTION
C  P[Y<=T] FOR A RANDOM VARIABLE Y HAVING A
C  STUDENT'S T DISTRIBUTION WITH N DEGREES OF
C  FREEDOM USING THE FUNCTION DFSHER.
C
      DOUBLE PRECISION DFSHER,T,F,ONE
      DATA ONE/1.DØ/
      I=1
      IF(T.LT.Ø.DØ) I=-1
      F=DFSHER(T*T,1,N)
      DSTDNT=(1+I*F)/2
      IF(F.EQ.-ONE) DSTDNT=-ONE
      RETURN
      END
```

BASIC Listing 1.

```
23ØØØ REM   SUBROUTINE TO CUMPUTE THE CUMULATIVE
23ØØ1 REM   DISTRIBUTION FUNCTION P[Y<=X] OF A
23ØØ2 REM   RANDOM VARIABLE Y HAVING A STANDARD
23ØØ3 REM   NORMAL DISTRIBUTION.  IT USES THE
23ØØ4 REM   VARIABLES X,Y,S,I,P.  THE PROBABILITY
23ØØ5 REM   COMPUTED IS DENOTED BY P.
23ØØ6 REM
23ØØ7 LET Y=X/SQR(2)
23ØØ8 IF X>Ø THEN 23Ø1Ø
23ØØ9 LET Y=-Y
23Ø1Ø LET S=Ø
23Ø11 FOR I=1 TO 37
23Ø12 LET S=S+EXP(-I*I/25)/I*SIN(2*I*Y/5)
23Ø13 NEXT I
23Ø14 LET S=S+Y/5
23Ø15 LET S=2*S/3.141592653589793
23Ø16 LET P=(1+S)/2
23Ø17 IF X>=Ø THEN 23Ø19
23Ø18 LET P=(1-S)/2
23Ø19 IF X<=8.3 THEN 23Ø21
23Ø2Ø LET P=1
23Ø21 IF X>=-8.3 THEN 23Ø23
23Ø22 LET P=Ø
23Ø23 RETURN
```

BASIC Listing 2.

```
22ØØØ REM   SUBROUTINE TO COMPUTE THE CUMULATIVE
22ØØ1 REM   DISTRIBUTION FUNCTION P[Y<=C] OF A
22ØØ2 REM   RANDOM VARIABLE Y HAVING A CHI SQUARE
22ØØ3 REM   DISTRIBUTION WITH N DEGREES OF FREEDOM.
22ØØ4 REM   IT USES THE SUBROUTINE FOR THE NORMAL
22ØØ5 REM   DISTRIBUTION AND THE VARIABLES C,C1,C2,
22ØØ6 REM   C3,C4,C5,C6,C7,C8,C9,P,P1,N,S$,T$.  THE
22ØØ7 REM   PROBABILITY COMPUTED IS DENOTED BY P.
22ØØ8 REM
22ØØ9 LET S$="THE CHI SQUARE SUBROUTINE "
22Ø1Ø LET T$="REQUIRES VALUES OF "
22Ø11 IF C>=Ø THEN 22Ø14
22Ø12 PRINT S$;T$;"C >= Ø"
22Ø13 GOTO 22Ø19
22Ø14 IF N>=1 THEN 22Ø17
22Ø15 PRINT S$;T$;"N >= 1"
22Ø16 GOTO 22Ø19
22Ø17 IF INT(N)=N THEN 22Ø21
22Ø18 PRINT S$;T$;"N TO BE INTEGER"
22Ø19 LET P=-1
22Ø2Ø RETURN
```

```
22021 IF C>0 THEN 22024
22022 LET P=0
22023 RETURN
22024 LET P1=3.141592653589793
22025 LET C1=0.5*C
22026 LET C2=INT(2*INT(N/2)/N)
22027 REM
22028 REM   L IS THE LARGEST VALUE OF X SUCH THAT
22029 REM   EXP(-X/2) IS ACCURATE FOR YOUR PARTICULAR
22030 REM   MACHINE.
22031 REM
22032 LET L=174.99646
22033 LET C3=0
22034 IF C<=L THEN 22036
22035 LET C3=1
22036 LET C9=0
22037 IF N<=2 THEN 22039
22038 LET C9=1
22039 LET C4=0
22040 IF C3=1 THEN 22043
22041 IF N=1 THEN 22043
22042 LET C4=EXP(-C1)
22043 LET C5=C4
22044 IF C2=1 THEN 22049
22045 LET X=-SQR(C)
22046 GOSUB 23000
22047 LET C5=2*P
22048 IF N=1 THEN 22061
22049 LET C6=0.5*(N-1)
22050 LET C7=(C2+1)/2
22051 IF C3=0 THEN 22063
22052 LET C8=0
22053 IF C2<>0 THEN 22055
22054 LET C8=LOG(SQR(P1))
22055 LET C9=LOG(C1)
22056 IF C7>C6 THEN 22061
22057 LET C8=C8+LOG(C7)
22058 LET C5=C5+EXP(C9*C7-C1-C8)
22059 LET C7=C7+1
22060 GOTO 22056
22061 LET P=1-C5
22062 RETURN
22063 LET C8=1
22064 IF C2<>0 THEN 22066
22065 LET C8=1/SQR(P1*C1)
22066 LET C9=0
22067 IF C7>C6 THEN 22072
22068 LET C8=C8*C1/C7
22069 LET C9=C9+C8
22070 LET C7=C7+1
22071 GOTO 22067
22072 LET P=1-C9*C4-C5
22073 RETURN
```

BASIC Listing 3.

```
25000 REM   SUBROUTINE TO COMPUTE THE CUMULATIVE
25001 REM   DISTRIBUTION FUNCTION P[Y<=F] OF A
25002 REM   RANDOM VARIABLE Y HAVING AN F
25003 REM   DISTRIBUTION WITH NUMERATOR DEGREES
25004 REM   OF FREEDOM N1 AND DENOMINATOR DEGREES
25005 REM   OF FREEDOM N2.  IT USES THE VARIABLES
25006 REM   F,F1,F2,F3,F4,F5,F6,F7,F8,F9,P,N1,N2,S$.
25007 REM   THE PROBABILITY COMPUTED IS DENOTED BY P.
25008 REM
25009 LET S$="THE F SUBROUTINE REQUIRES VALUES OF "
25010 IF F>=0 THEN 25013
25011 PRINT S$;"F >= 0"
25012 GOTO 25024
25013 IF INT(N1)=N1 THEN 25016
25014 PRINT S$;"N1 TO BE INTEGER"
25015 GOTO 25024
25016 IF INT(N2)=N2 THEN 25019
25017 PRINT S$;"N2 TO BE INTEGER"
25018 GOTO 25024
25019 IF N1>=1 THEN 25022
25020 PRINT S$;"N1 >= 1"
25021 GOTO 25024
25022 IF N2>=1 THEN 25026
25023 PRINT S$;"N2 >= 1"
25024 LET P=-1
25025 RETURN
25026 IF F=0 THEN 25077
25027 LET F1=2*INT(N1/2)-N1+2
25028 LET F2=2*INT(N2/2)-N2+2
25029 LET F3=F*N1/N2
25030 LET F4=1/(1+F3)
25031 IF F1<>1 THEN 25046
25032 IF F2<>1 THEN 25043
25033 LET P=SQR(F3)
25034 REM
25035 REM   F7 IS 1/PI.  IT MUST BE ENTERED WITH
25036 REM   K+1 DECIMAL PLACES TO GUARANTEE K
25037 REM   DECIMAL PLACE ACCURACY IN THE CDF.
25038 REM
25039 LET F7=.31830988618379067
25040 LET F6=F7*F4/P
25041 LET P=2*F7*ATN(P)
25042 GOTO 25053
25043 LET P=SQR(F3*F4)
25044 LET F6=0.5*P*F4/F3
25045 GOTO 25053
25046 IF F2<>1 THEN 25051
25047 LET P=SQR(F4)
25048 LET F6=0.5*F4*P
25049 LET P=1-P
25050 GOTO 25053
25051 LET F6=F4*F4
25052 LET P=F3*F4
25053 LET F7=2*F3/F4
25054 IF F1<>1 THEN 25061
25055 LET F5=F2+2
25056 IF F5>N2 THEN 25064
25057 LET F6=(1+F1/(F5-2))*F6*F4
25058 LET P=P+F6*F7/(F5-1)
25059 LET F5=F5+2
25060 GOTO 25056
25061 LET F8=F4^INT((N2-1)/2)
25062 LET F6=F6*F8*N2/F2
25063 LET P=P*F8+F3*F4*(F8-1)/(F4-1)
25064 LET F7=F3*F4
25065 LET F4=2/F4
25066 LET F2=N2-2
25067 LET F9=F1+2
25068 IF F9>N1 THEN 25074
25069 LET F5=F9+F2
25070 LET F6=F7*F6*F5/(F9-2)
25071 LET P=P-F4*F6/F5
25072 LET F9=F9+2
25073 GOTO 25068
25074 IF P<1 THEN 25076
25075 LET P=1
25076 IF P>0 THEN 25078
25077 LET P=0
25078 RETURN
```

BASIC Listing 4.

```
21000 REM   SUBROUTINE TO COMPUTE THE CUMULATIVE
21001 REM   DISTRIBUTION FUNCTION P[Y<=T] OF A
21002 REM   RANDOM VARIABLE Y HAVING A STUDENT'S
21003 REM   T DISTRIBUTION WITH N DEGREES OF
21004 REM   FREEDOM.  IT USES THE SUBROUTINE FOR THE
21005 REM   F DISTRIDUTION AND THE VARIABLES T,N,P.
21006 REM   THE PROBABILITY COMPUTED IS DENOTED BY P.
21007 REM
21008 LET F=T*T
21009 LET N1=1
21010 LET N2=N
21011 GOSUB 25000
21012 IF P=-1 THEN 21014
21013 LET P=0.5*(1+SGN(T)*P)
21014 RETURN
25000 REM
25001 REM   SUBROUTINE TO COMPUTE F CDF GOES HERE
25002 REM
```