บรรณานุกรม

หนังสือ

Deo, Narsigh. System Simulation with Digital Computer. India: Prentice-Hall, 1979.

Graybill, Franklin A. Theory and Application of the Linear Model. U.S.A. : Duxbury Press, 1976.

IBM Application Program System/360 Scientific Subroutine Package (360. A-ex. 03x) Version III.

Rubinstein, Reuwen Y. Simulation and Monte Carlo Method. New York: John Wiley & Son, 1981.

Searle, S.R. Linear Models. U.S.A. : John Wiley & Son, 1971.

วารสาร

Corbell, R.R. ; Searle, S.R. "Restricted Maximum Likelihood (REML) Estimation of Variance Component in the Mixed Mode. " Technometrics 18 (February 1976) : 31-38.

Harville, David A. "Maximum Likelihood Approachs to Variance Component Estimation and to Related Problems." Journal of the American Statistical Association 72 (June 1977): 320-340.

Jennrich, R.I. ; Sampson, P.F. "Newton-Raphson and Related Algorithms for Maximum Likelihood Variance Component Estimation." Technometrics 18 (February 1976) : 11-17.

Marsaglia, G.; Bray, T.A. "A Convenient method for generating Normal Variables." SIAM Review 6 (1964) : 260-264

Miller, John J. "Maximum Likelihood etimation of Variance Companents-A Monte Carlo Study." Journal of Slatistical Computation and Simulation 8 (1979) : 175-190.

Rao, C. Radhakrishna. "Estimation of Variance and Covariance Components-MINQUE Theory." Journal of Multivariate Analysis 1 (1971) : 257-275.

————. "Estimation of Variance and Covariance Components in Linear Models." Journal of American Statistical Association 67 (March 1972) : 112-115

————. "Minimum Variance Quadratic Unbiased Estimation of Variance Components." Journal of Multivariate Analysis 1 (1971): 445-465.

Swallow, William H. "Variances of Locally Minimum Variance Quadratic Unbiased Estimators ("MIVQUE'S") of Variance Components." Technometrics 23 (August 1981): 271-283.

————. Monahan, Johm F. "Monte Carlo Comparison of ANOVA, MIVQUE, REML and ML Estimators of Variance Components." Thechnometrics 26 (February 1984): 47-57.

————. Searle, S.R. "Minimum Variance Quadratic Unbiased Estimation (MIVQUE) of Variance Component." Technometrics 20 (August 1978): 265-272.

วิทยานิพนธ์

ลุกัลยา ธรรมรักษา. "การศึกษาเปรียบเทียบวิธีการประมาณค่าแวเรียนซ์คอมโพเนนท์ ในแผนแบบไม่สมดุลย์ในกรณีการแจกแจงแบบสองทาง." วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาสถิติ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2524.

ภาคผนวก

```
C.....
C....                                                                  ----
C....                    PROGRAM PROCESS                               ----
C....                    ===============                               ----
C....                                                                  ----
C....      1. GENERATE RANDOM NUMBER WITH UNIFORM DISTRIBUTION         ----
C....      2. GENERATE DATA WITH NORMAL DISTRIBUTION                   ----
C....           MEAN                    =    ZERO                     ----
C....           STANDARD DIVIATION      =    SIGMA                    ----
C....      3. GENERATE DATA WITH 2-WAY CLASSIFICATION                  ----
C....         AND UNBALANCED DESIGN                                    ----
C....         RANDOM MODEL:                                            ----
C....                                                                  ----
C....                    Y    = U + A  +  B  +  AB   + E              ----
C....                     IJK        I     J      IJ     IJK          ----
C....                                                                  ----
C....      4. ESTIMATE VARIANCE COMPONENT BY 4 METHOD :                ----
C....           4.1  ANOVA METHOD                                     ----
C....           4.2  ML METHOD WITH ANOVA AS PRIOR                    ----
C....           4.3  MIVQUE METHOD WITH ANOVA AS PRIOR                ----
C....           4.4  I-MIVQUE METHOD  BY ITERATING MIVQUE            ----
C....      5. CALCULATE MSE(ESTIMATOR) OF EACH METHOD IN 4.            ----
C....                                                                  ----
C....                                                                  ----
C.....
C                                                                      C
C                    VARIABLES  DESCRIPTION                            C
C                    =====================                             C
C                                                                      C
C.....C
C                                                                      C
C   VARIABLES                          DESCRIPTION                     C
C   =========                          ===========                     C
C                                                                      C
C   ISEED1          SEED FOR GENERATE A-EFFECT    ==>   973253        C
C   ISEED2          SEED FOR GENERATE B-EFFECT    ==>   876541        C
C   ISEED3          SEED FOR GENERATE AB-EFFECT   ==>   778467        C
C   ISEED4          SEED FOR GENERATE E-EFFECT    ==>   392485        C
C   MEANS           MEAN OF EACH EFFECT           ==>   0             C
C   MEAN            GRAND MEAN                                        C
C   KOUNT1          NO. OF REPEAT SIMULATION = 200 TIMES              C
C   SMSE(I)         SUM OF MSE(ESTIMATOR) SQUARE IN I-TH METHOD       C
C   MSE(I,J)        MEAN SQUARE ERROR OF J-TH VARIANCE COMPONENT      C
C                   ESTIMATOR IN THE I-TH METHOD                      C
C   SS(I,J)         SUMMATION OF (ESTIMATOR-PARAMETOR) * * 2          C
C                   IN THE J-TH COMPONENT AND I-TH METHOD             C
C   LI              NO. LEVEL OF FACTOR  A                            C
C   LJ              NO. LEVEL OF FACTOR  B                            C
C   NC(I,J)         NO. OF OBSERVATION IN CELL (I,J)                  C
C   SIGMA(I)        STANDARD DEVIATION OF THE I-TH COMPONENT          C
C   U1(I,J)         I*J MATRIX OF COEFFICIENT OF A-EFFECT             C
C   U2(I,J)         I*J MATRIX OF COEFFICIENT OF B-EFFECT             C
C   U3(I,J)         I*J MATRIX OF COEFFICIENT OF AB-EFFECT            C
C   V1(I,J)         I*J MATRIX OF U1*U1-TRANSPOST                     C
C   V2(I,J)         I*J MATRIX OF U2*U2-TRANSPOST                     C
C   V3(I,J)         I*J MATRIX OF U3*U3-TRANSPOST                     C
C   V4(I,J)         IDENTITY MATRIX                                   C
C   V(I,J)          I*J MATRIX OF SUMMATION OF SIGMA SQUARE *         C
C                       U * U-TRANSPOST                               C
C   VE(I,J)         I*J MATRIX OF VARIANCE J-TH COMPONENT ESTIMATOR   C
C                       OF I-TH METHOD                                C
C   VIN(I,J)        I*J MATRIX OF V-INVERT                            C
C   X(I)            VECTOR OF 1                                       C
C   Y(I)            VECTOR OF OBSERVATION                             C
C   Y(I,J,K)        OBSERVATION VALUE AT THE I-TH LEVEL OF A-FACTOR ,  C
C                   J-TH LEVEL OF B-FACTOR AND K-TH VALUE OF CELL (I,J) C
C   ALPHA(I)        RANDOM EFFECT OF THE I-TH LEVEL OF FACTOR A       C
C   BETA(J)         RANDOM EFFECT OF THE J-TH LEVEL OF FACTOR B       C
C   DINTRC(JI)      RANDOM EFFECT OF THE I-TH,J-TH INTERACTION AB     C
C   ERROR(I)        RANDOM EFFECT OF ERROR                            C
C   EFFECTA(I)      EFFECT OF FACTOR A IN THE I-TH LEVEL              C
C   EFFECTB(J)      EFFECT OF FACTOR B IN THE J-TH LEVEL              C
C   DINTRE(JI)      EFFECT OF INTERACTION AB                          C
C   C(I)            DUMMY VECTOR                                      C
C   F(I)              "                                               C
C   F(I)              "                                               C
C   IROW(I)           "                                               C
C   ICOL(I)           "                                               C
```

```
C                                                                       C
C.....................................................................C
C                                                                       C
C   SUBROUTINE                                                          C
C   ==========                                                          C
C                                                                       C
C   MINV            SUBROUTINE FROM IBM USER GUIDE TO FIND INVERT       C
C                   MATRIX AND DETERMONANT                              C
C   GMPRD           SUBROUTINE FROM IBM USER GUIDE FOR MUTIPLY MATRIX   C
C   MFGR            SUBROUTINE FROM IBM USER GUIDE TO FIND RANK OF      C
C                   MATRIX                                              C
C   TRACE           SUBROUTINE TO FIND TRACE OF MATRIX                  C
C   RANDU           SUBROUTINE TO GENERATE RANDOM NUMBER WITH UNIFORM   C
C                   DISTRIBUTION                                        C
C                                                                       C
C.....................................................................C
C                                                                       C
C   FUNCTION                                                            C
C   ========                                                            C
C                                                                       C
C   NORM            FOR GENERATING DATA WITH NORMAL DISTRIBUTION        C
C                                                                       C
C.....................................................................C
C
      DIMENSION NC(6,6),U1(30,6),U2(30,6),L3(30,36),V1(30,30),
     1    V2(30,30),V3(30,30),V4(30,30),V(30,30),Y(30),X(30),
     2    VIN(30,30),YSX(30),VV1(30,30),VV2(30,30),VV3(30,30),
     3    VV4(30,30),VE(4,4),S(4),TR(4,4),IROW(30),ICOL(30),C(900),
     4    E(900),F(900),VJV1(30,30),VR(4),C(30,30),XVXI(1),VA(4,4),
     5    SS(4,4),SIGMA(4),SIGMAS(4),ALPHA(6),BETA(6),DINTRC(6),
     6    ERROR(30),EFECTA(30),EFECTB(30),DINTER(30),Y1(6,6,30),
     7    TTA(6),TTB(6),TTAB(6,6),NI(6),NJ(6),P(3,3),SMSE(4)
C
C.....................................................................C
C
      REAL MSE(4,4),MEAN,NORM,LIKH,MAX
C.....................................................................C
C
      ISEED1 = 973253
      ISEED2 = 876541
      ISEED3 = 778467
      ISEED4 = 392485
      MEANS = 0
      MEAN = 0
      KOUNT1 = 0
      DO 111 I = 1,4
      SMSE(I) = 0
      DO 111 J = 1,4
  111 SS(I,J) = 0
      READ(5,1114) (SIGMAS(I),I=1,4),CV
      READ(5,1112) LI,LJ
      READ(5,1112) ((NC(I,J),J=1,LJ),I=1,LI)
 1112 FORMAT(24I3,/,24I3)
 1114 FORMAT(4(F3.0,1X),F3.1)
      DO 103 I =1,4
      MEAN = MEAN + SIGMAS(I)
  103 SIGMA(I) = SQRT(SIGMAS(I))
      MEAN = SQRT(MEAN) / CV
C
C  U1 MATRIX
C
      DO 1002 L = 1,LI
      N= 0
      DO 1004 I = 1,LI
      DO 1004 J = 1,LJ
      N1 = NC(I,J)
      IF (I .NE. L) GO TO 1008
      DO 1006 K = 1,N1
      N = N + 1
 1006 U1(N,L) = 1
      GO TO 1004
 1008 DO 1010 K = 1,N1
      N = N + 1
 1010 U1(N,L) = 0
 1004 CONTINUE
 1002 CONTINUE
      K = 0
      DO 1040 I = 1,LI
      DO 1040 J = 1,N
      K = K + 1
 1040 C(K) = U1(J,I)
```

```
      K = 0
      DO 1043 J = 1,N
      DO 1043 I = 1,LI
      K = K + 1
1043  F(K) = U1(J,I)
      CALL GMPRD(C,F,E,N,LI,N)
      K = 0
      DO 1042 J = 1,N
      DO 1042 I = 1,N
      K = K + 1
1042  V1(I,J) = E(K)
C
C U2 MATRIX
C
      DO 1012 L = 1,LJ
      N = 0
      DO 1014 I = 1,LI
      DO 1014 J = 1,LJ
      N1 = NC(I,J)
      IF ( J .NE. L ) GO TO 1018
      DO 1016 K = 1,N1
      N = N + 1
1016  U2(N,L) = 1
      GO TO 1014
1018  DO 1020 K = 1,N1
      N = N + 1
1020  U2(N,L) = 0
1014  CONTINUE
1012  CONTINUE
      K = 0
      DO 1044 J = 1,LJ
      DO 1044 I = 1,N
      K = K + 1
1044  C(K) = U2(I,J)
      K = 0
      DO 1045 I = 1,N
      DO 1045 J = 1,LJ
      K = K + 1
1045  F(K) = U2(I,J)
      CALL GMPRD(C,F,E,N,LJ,N)
      K = 0
      DO 1046 J = 1,N
      DO 1046 I = 1,N
      K = K + 1
1046  V2(I,J) = E(K)
C
C U3 MATRIX
C
      JI = LI* LJ
      DO 1022 L = 1,JI
      N = 0
      IC = 0
      DO 1024 I = 1,LI
      DO 1024 J = 1,LJ
      IC = IC + 1
      N1 = NC(I,J)
      IF ( IC .NE. L) GO TO 1028
      DO 1026 K = 1,N1
      N = N + 1
1026  U3(N,L) = 1
      GO TO 1024
1028  DO 1030 K = 1,N1
      N = N + 1
1030  U3(N,L) = 0
1024  CONTINUE
1022  CONTINUE
      K = 0
      DO 1048 J = 1,JI
      DO 1048 I = 1,N
      K = K + 1
1048  C(K) = U3(I,J)
      K = 0
      DO 1049 I = 1,N
      DO 1049 J = 1,JI
      K = K + 1
1049  F(K) = U3(I,J)
      CALL GMPRD(C,F,E,N,JI,N)
      K = 0
      DO 1050 J = 1,N
      DO 1050 I = 1,N
      K = K + 1
1050  V3(I,J) = E(K)
```

```
C
C V4 MATRIX
C
      DO 1032 I = 1,N
      DO 1032 J = 1,N
      IF ( I - J ) 1034,1036,1034
1034  V4(I,J) = 0
      GO TO 1032
1036  V4(I,J) = 1
1032  CONTINUE
C
C X MATRIX
C
      DO 1052 I = 1,N
1052  X(I) = 1
C
      WRITE(6,7777)
7777  FORMAT(T15,'ANOVA METHOD',T45,'MLE METHOD',T75,'MIVQUE METHOD',
     1 T105,'I - MIVQUE METHOD',/)
      F1 = - N / 2.0
      F2 = 6.28318 * * ( F1 )
  99  DO 14 I = 1,LI
C
C  GENERATE A-EFFECT
C
      ALPHA(I) = NORM( ISEED1,SIGMA(1),MEANS )
  14  CONTINUE
C
C  GENERATE B-EFFECT
C
      DO 16 J = 1,LJ
      BETA(J) = NORM( ISEED2,SIGMA(2),MEANS )
  16  CONTINUE
C
C  GENERATE INTERACTION AB-EFFECT
C
      JI = 0
      DO 18 J = 1,LJ
      DO 20 I = 1,LI
      JI = JI + 1
      DINTRC(JI) = NORM( ISEED3,SIGMA(3),MEANS )
  20  CONTINUE
  18  CONTINUE
C
C  GENERATE ERROR-EFFECT
C
      N = 0
      DO 24 I = 1,LI
      DO 26 J = 1,LJ
      N1 = NC(I,J)
      DO 28 K = 1,N1
      N = N + 1
      ERROR(N) = NORM( ISEED4,SIGMA(4),MEANS )
  28  CONTINUE
  26  CONTINUE
  24  CONTINUE
C
C  INITIALIZE VALUE
C
      DO 153 I = 1,N
      EFECTA(I) = 0
      EFECTB(I) = 0
      DINTER(I) = 0
 153  CONTINUE
C
C  MULTIPLY U1 BY ALPHA
C
      DO 92 I = 1,N
      DO 92 L = 1,LI
  92  EFECTA(I) = EFECTA(I) + ( U1(I,L)*ALPHA(L) )
C
C   MULTIPLY U2 BY BETA
C
      DO 113 I = 1,N
      DO 113 L = 1,LI
 113  EFECTB(I) = EFECTB(I) + ( U2(I,L) * BETA(L) )
C
C   MULTIPLY U3 BY INTRAC
C
      DO 123 I = 1,N
      DO 123 L = 1,JI
 123  DINTER(I) = DINTER(I) + ( U3(I,L) * DINTRC(L) )
```

```
C
C     ADDITION MATRIX TO GET Y(N) OR Y(I,J,K)
C
      DO 133 I = 1,N
  133 Y(I) = MEAN + EFECTA(I) + EFECTB(I) + DINTER(I) + ERROR(I)
      KK = 0
      DO 202 I = 1,LI
      DO 202 J = 1,LJ
      N1 = NC(I,J)
      DO 202 K = 1,N1
      KK = KK + 1
  202 Y1(I,J,K) = Y(KK)
C
C.....                                                                C
C                                                                     C
C          =========>   THE ANOVA METHOD   <=========                 C
C                                                                     C
C.....                                                                C
C
C    FIND TO
C
      TO = 0
      DO 210 I = 1,LI
      DO 210 J = 1,LJ
      N1 = NC(I,J)
      DO 210 K = 1,N1
  210  TO = TO + Y1(I,J,K) ** 2
C
C    FIND TA
C
      TA = 0
      DO 213 I = 1,LI
      TTA(I) = 0
      NI(I) = 0
      DO 214 J = 1,LJ
      NI(I) = NI(I) + NC(I,J)
      N1 = NC(I,J)
      DO 217 K = 1,N1
  217  TTA(I) = TTA(I) + Y1(I,J,K)
  214  CONTINUE
  213  TA = TA + ( TTA(I) * * 2 ) / NI(I)
C
C    FIND TB
C
      TB = 0
      DO 219 J = 1,LJ
      TTB(J) = 0
      NJ(J) = 0
      DO 221 I = 1,LI
      NJ(J) = NJ(J) + NC(I,J)
      N1 = NC(I,J)
      DO 223 K = 1,N1
  223  TTB(J) = TTB(J) + Y1(I,J,K)
  221  CONTINUE
  219  TB = TB + (TTB(J) ** 2) / NJ(J)
C
C   FIND TAB
C
      TAB = 0
      DO 224 I = 1,LI
      DO 224 J = 1,LJ
      TTAB(I,J) = 0
      N1 = NC(I,J)
      DO 226 K = 1,N1
  226  TTAB(I,J) = TTAB(I,J) + Y1(I,J,K)
  224  TAB = TAB + ( TTAB(I,J) * * 2 ) / NC(I,J)
C
C    FIND TU
C
      TTU = 0
      DO 228 I = 1,LI
      DO 228 J = 1,LJ
      N1 = NC(I,J)
      DO 235 K = 1,N1
  235  TTU = TTU + Y1(I,J,K)
  228  TU = (TTU * * 2) / N
```

```
C
C    FIND K1,K2,K3,K4,K23
C
       AK1 = 0
       AK2 = 0
       AK3 = 0
       AK4 = 0
       AK23 = 0
C
C    FIND K1
C
       DO 232 I = 1,LI
232    AK1 = AK1 + NI(I) * * 2
C
C    FIND K2
C
       DO 234 J = 1,LJ
234     AK2 = AK2 + NJ(J) * * 2
C
C   FIND K3 & K23
C
       DO 236 I = 1,LI
       DUM = 0
       DO 238 J = 1,LJ
238     DUM = DUM + NC(I,J) * * 2
        AK23 = AK23 + DUM
236     AK3 = AK3 + DUM / NI(I)
C
C    FIND  K4
C
       DO 240 J = 1,LJ
       DUM = 0
       DO 242 I = 1,LI
242     DUM = DUM + NC(I,J) * * 2
240     AK4 = AK4 + DUM / NJ(J)
       S1 = LI * LJ
       VE(1,4) = ( TO - TAB ) / ( N - S1 )
C
C    FIND MATRIX P
C
       P(1,1) = N - ( AK1 / N )
       P(1,2) = AK3 - ( AK2 / N )
       P(1,3) = AK3 - ( AK23 / N )
       P(2,1) = AK4 - ( AK1 / N )
       P(2,2) = N - ( AK2 / N )
       P(2,3) = AK4 - ( AK23 / N )
       P(3,1) = ( AK1 / N ) - AK4
       P(3,2) = ( AK2 / N ) - AK3
       P(3,3) = N - AK3 - AK4 + ( AK23 / N )
C
C    FIND P INVERT
C
       KK = 0
       DO 303 J = 1,3
       DO 303 I = 1,3
       KK = KK + 1
 303   C(KK) = P(I,J)
       CALL MINV( C,3,D,IROW,ICOL )
       KK = 0
       DO 307 I = 1,3
       DO 307 J = 1,3
       KK = KK + 1
 307   P(I,J) = C(KK)
C
C    FIND OTHER MATRIX
C
       S(1) = TA - TO - ( LI - 1 ) * VE(1,4)
       S(2) = TB - TO - ( LJ - 1 ) * VE(1,4)
       S(3) = TAB - TA - TB + TO - ( S1 - LI - LJ + 1 ) * VE(1,4)
C
C    FIND VARIANCE COMPONENT
C
       DO 244 I = 1,3
       DUM = 0
       DO 246 J = 1,3
246     DUM = DUM + P(I,J) * S(J)
244     VE(1,I) = DUM
       DO 260 I = 1,4
       IF ( VE(1,I) .LE. 0 ) VE(1,I) = 0
260    CONTINUE
```

```
C.....                                                                  C
C                                                                       C
C       ========>   THE MAXIMUM LIKELIHOOD METHOD   <========          C
C                                                                       C
C.....                                                                  C
C
      DO 101 J = 1,4
 101  VR(J) = VE(1,J)
      ND = 0
      INDEX = 0
      IA = 0
      EPS = 1.0E-7
2     DO 3 I=1,N
      DO 3 J=1,N
3     V(I,J) = VR(1)*V1(I,J)+VR(2)*V2(I,J)+VR(3)*V3(I,J)+ VR(4)*V4(I,J)
      K = 0
      DO 4 I=1,N
      DO 4 J=1,N
      K = K + 1
      F(K) = V(J,I)
4     C(K) = V(J,I)
      CALL MFGR(C,N,N,EPS,IRANK,IROW,ICOL)
      IF(N - IRANK) 5,6,5
5     WRITE(6,55)
55    FORMAT( T5,'RANK OF MATRIX IS NOT N' )
      GO TO  99
6     CALL MINV(F,N,D,IROW,ICOL)
      IF ( INDEX .EQ. 0 )  GO TO 90
      IF ( D .GT. 0) GO TO 450
      WRITE(6,452)
 452  FORMAT(/,T5,'DETERMINANT < 0' )
      GO TO 99
 450  CALL GMPRD(F,YSX,C,N,N,1)
      CALL GMPRD(YSX,C,F,1,N,1)
      FACT2 = EXP( - F(1) / 2 )
      FACT1 = F2  / SQRT(D)
      LIKH = FACT1 * FACT2
      IA = IA + 1
      INDEX = 0
      GO TO 220
 90   K = 0
      DO 100 I = 1,N
      DO 100 J = 1,N
      K = K + 1
 100  VIN(I,J) = F(K)
C
C ( X-TRANSPOST * V-INVERT ) 1*N
C
      CALL GMPRD(X,F,C,1,N,N)
C
C ( X-TRANSPOST * V-INVERT * Y ) 1*1
C
      CALL GMPRD(C,Y,E,1,N,1)
C
C ( X-TRANSPOST * V-INVERT * X )1*1
C
      CALL GMPRD(C,X,F,1,N,1)
C
      BET  = E(1) / F(1)
      DO 102 I = 1,N
 102  YSX(I) = Y(I) - X(I) * BET
      K = 0
      DO 104 J = 1,N
      DO 104 I = 1,N
      K = K + 1
 104  C(K) = VIN(I,J)
      DO 142 II = 1,4
      GO TO ( 106,110,114,118 ) , II
 106  K = 0
      DO 108 J = 1,N
      DO 108 I = 1,N
      K = K + 1
 108  F(K) = V1(I,J)
      GO TO 122
```

```
 110  K = 0
      DO 112 J = 1,N
      DO 112 I = 1,N
      K = K + 1
 112  F(K) = V2(I,J)
      GO TO 122
 114  K = 0
      DO 116 J = 1,N
      DO 116 I = 1,N
      K = K + 1
 116  F(K) = V3(I,J)
      GO TO 122
 118  K = 0
      DO 120 J = 1,N
      DO 120 I = 1,N
      K = K + 1
 120  F(K) = V4(I,J)
C
C ( V-INVERT * V(I) ) N*N
C
 122  CALL GMPRD(C,F,E,N,N,N)
      GO TO (124,128,132,136) , II
 124  K = 0
      DO 126 J = 1,N
      DO 126 I = 1,N
      K = K + 1
 126  VV1(I,J) = E(K)
      GO TO 140
 128  K = 0
      DO 130 J = 1,N
      DO 130 I = 1,N
      K = K + 1
 130  VV2(I,J) = E(K)
      GO TO 140
 132  K = 0
      DO 134 J = 1,N
      DO 134 I = 1,N
      K = K + 1
 134  VV3(I,J) = E(K)
      GO TO 140
 136  K = 0
      DO 138 J = 1,N
      DO 138 I = 1,N
      K = K + 1
 138  VV4(I,J) = E(K)
C
C ( V-INVERT * V(I) * V-INERT ) N*N
C
 140  CALL GMPRD(E,C,F,N,N,N)
C
C ( (Y-XBETA)-TRANSPOST * V-INVERT * V(I) * V-INVERT ) 1*N
C
      CALL GMPRD(YSX,F,E,1,N,N)
C
C ( (Y-XBETA)-TRANSPOST * V-INVERT * V(I) * V-INVERT * (Y-XBETA) ) 1*1
C
      CALL GMPRD(E,YSX,F,1,N,1)
      S(II) = F(1)
 142  CONTINUE
C
C TRANSFORM V(J) TO F
C
      DO 184 JJ = 1,4
      GO TO ( 144,148,152,156 ) , JJ
 144  K = 0
      DO 146 J = 1,N
      DO 146 I = 1,N
      K = K + 1
 146  F(K) = VV1(I,J)
      GO TO 160
 148  K = 0
      DO 150 J = 1,N
      DO 150 I = 1,N
      K = K + 1
 150  F(K) = VV2(I,J)
      GO TO 160
 152  K = 0
      DO 154 J = 1,N
      DO 154 I = 1,N
      K = K + 1
 154  F(K) = VV3(I,J)
      GO TO 160
```

```
 156  K = 0
      DO 158 J = 1,N
      DO 158 I = 1,N
      K = K + 1
 158  F(K) = VV4(I,J)
C
C TRANSFORM V(I) TO C
C
 160  DO 182 II = 1,4
      GO TO ( 162,166,170,174 ) , II
 162  K = 0
      DO 164 J = 1,N
      DO 164 I = 1,N
      K = K + 1
 164  C(K) = VV1(I,J)
      GO TO 178
 166  K = 0
      DO 168 J = 1,N
      DO 168 I = 1,N
      K = K + 1
 168  C(K) = VV2(I,J)
      GO TO 178
 170  K = 0
      DO 172 J = 1,N
      DO 172 I = 1,N
      K = K + 1
 172  C(K) = VV3(I,J)
      GO TO 178
 174  K = 0
      DO 176 J = 1,N
      DO 176 I = 1,N
      K = K + 1
 176  C(K) = VV4(I,J)
C
C ( V-INVERT * V(J) * V-INVERT * V(I) ) N*N
C
 178  CALL GMPRD(C,F,E,N,N,N)
      K = 0
      DO 180 J = 1,N
      DO 180 I = 1,N
      K = K + 1
 180  VJVI(I,J) = E(K)
C
C TR( V-INVERT * V(J) * V-INVERT * V(I) )
C
      CALL TRACE(VJVI,N,SUM)
      TR(JJ,II) = SUM
 182  CONTINUE
 184  CONTINUE
C
      K = 0
      DO 196 J = 1,4
      F(J) = S(J)
      DO 194 I = 1,4
      K = K + 1
      E(K) = TR(I,J)
 194  C(K) = TR(I,J)
 196  CONTINUE
      L = 4
      K = 0
      CALL MFGR(E,L,L,EPS,IRANK,IROW,ICOL)
      IF ( IRANK - L ) 231,200,231
 231  WRITE(6,1111) IRANK
1111  FORMAT(T5,'RANK = ',I4 )
      GO TO 99
 200  CALL MINV(C,L,D,IROW,ICOL)
      CALL GMPRD(C,F,E,L,L,1)
      DO 212 J = 1,L
 212  VR(J) = E(J)
      DO 216 J = 1,4
      IF ( VR(J) .LT. 0 ) VR(J) = 0
 216  CONTINUE
      INDEX = 1
 220  IF ( INDEX .EQ. 1 ) GO TO 2
      IF ( IA .EQ. L ) GO TO 454
      IF ( MAX .GE. LIKH ) GO TO 456
 454  MAX = LIKH
C
C  VARIANCE COMPONENT
C
      DO 458 J = 1,4
 458  VE(2,J) = VR(J)
      IE = 0
      GO TO 2
 456  IB = IB + 1
      IF ( IB  .LT. 2 ) GO TO 2
```

```
C                                                                       C
C.......................................................................C
C                                                                       C
C    ==========>   THE MIVQUE & THE I-MIVQUE METHOD   <==========      C
C                                                                       C
C.......................................................................C
C                                                                       C
      DO 300 J = 1,4
 300  VR(J) = VE(1,J)
      ND = 0
302   DO 304 I=1,N
      DO 304 J=1,N
304   V(I,J) = VR(1)*V1(I,J)+VR(2)*V2(I,J)+VR(3)*V3(I,J)+ VR(4)*V4(I,J)
      K = 0
      DO 306 I=1,N
      DO 306 J=1,N
      K = K + 1
      F(K) = V(J,I)
306   C(K) = V(J,I)
      CALL MFGR(C,N,N,EPS,IRANK,IROW,ICOL)
      IF(N - IRANK) 308,310,308
308   WRITE(6,305)
305   FORMAT( T5,'RANK OF MATRIX IS NOT N' )
      GO TO 99
310   CALL MINV(F,N,D,IROW,ICOL)
      K = 0
      DO 312 I = 1,N
      DO 312 J = 1,N
      K = K + 1
 312  VIN(I,J) = F(K)
C
C ( X-TRANSPOST * V-INVERT ) 1*N
C
      CALL GMPRD(X,F,C,1,N,N)
C
C ( V-INVERT * X ) N*1
C
      CALL GMPRD(F,X,E,N,N,1)
C
C ( X-TRANSPOST * V-INVERT * X ) 1*1
C
      CALL GMPRD(C,X,F,1,N,1)
C
C ( X-TRANSPOST * V-INVERT * X ) -INVERT
C
      XVXI(1) = 1 / F(1)
C
C ( V-INVERT * X * (X-T*V-I*X)-INVERT ) N*1
C
      CALL GMPRD(E,XVXI,F,N,1,1)
C
C ( V-INVERT * X * (X-T * V-I *X)-INVERT * X-TRANSPOST * V-INVERT
C
      CALL GMPRD(F,C,E,N,1,N)
C
C MATRIX Q (N*N)
C
      K = 0
      DO 314 J = 1,N
      DO 314 I = 1,N
      K = K + 1
 314  VJVI(I,J) = E(K)
      DO 316 I = 1,N
      DO 316 J = 1,N
 316  Q(I,J) = VIN(I,J) - VJVI(I,J)
C
C
      K = 0
      DO 318 J = 1,N
      DO 318 I = 1,N
      K = K + 1
 318  C(K) = Q(I,J)
      DO 356 II = 1,4
      GO TO ( 320,324,328,332 ) , II
 320  K = 0
      DO 322 J = 1,N
      DO 322 I = 1,N
      K = K + 1
 322  F(K) = V1(I,J)
      GO TO 336
```

```
 324  K = 0
      DO 326 J = 1,N
      DO 326 I = 1,N
      K = K + 1
 326  F(K) = V2(I,J)
      GO TO 336
 328  K = 0
      DO 330 J = 1,N
      DO 330 I = 1,N
      K = K + 1
 330  F(K) = V3(I,J)
      GO TO 336
 332  K = 0
      DO 334 J = 1,N
      DO 334 I = 1,N
      K = K + 1
 334  F(K) = V4(I,J)
C
C ( Q * V(I) ) N*N
C
 336  CALL GMPRD(C,F,E,N,N,N)
C
      GO TO (338,342,346,350) , II
 338  K = 0
      DO 340 J = 1,N
      DO 340 I = 1,N
      K = K + 1
 340  VV1(I,J) = E(K)
      GO TO 354
 342  K = 0
      DO 344 J = 1,N
      DO 344 I = 1,N
      K = K + 1
 344  VV2(I,J) = E(K)
      GO TO 354
 346  K = 0
      DO 348 J = 1,N
      DO 348 I = 1,N
      K = K + 1
 348  VV3(I,J) = E(K)
      GO TO 354
 350  K = 0
      DO 352 J = 1,N
      DO 352 I = 1,N
      K = K + 1
 352  VV4(I,J) = E(K)
C
C ( Q * V(I) * Q ) N*N
C
 354  CALL GMPRD(E,C,F,N,N,N)
C
C ( Y-TRANSPOST * Q * V(I) * C ) 1*N
C
      CALL GMPRD(Y,F,E,1,N,N)
C
C ( Y-TRANSPOST *  Q  * V(I) * Q * Y ) 1*1
C
      CALL GMPRD(E,Y,F,1,N,1)
      S(II) = F(1)
 356  CONTINUE
C
C TRANSFORM V(J) TO F
C
      DO 398 JJ = 1,4
      GO TO ( 358,362,366,370 ) , JJ
 358  K = 0
      DO 360 J = 1,N
      DO 360 I = 1,N
      K = K + 1
 360  F(K) = VV1(I,J)
      GO TO 374
 362  K = 0
      DO 364 J = 1,N
      DO 364 I = 1,N
      K = K + 1
 364  F(K) = VV2(I,J)
      GO TO 374
```

```
 366  K = 0
      DO 368 J = 1,N
      DO 368 I = 1,N
      K = K + 1
 368  F(K) = VV3(I,J)
      GO TO 374
 370  K = 0
      DO 372 J = 1,N
      DO 372 I = 1,N
      K = K + 1
 372  F(K) = VV4(I,J)
C
C TRANSFORM V(I) TO C
C
 374  DO 396 II = 1,4
      GO TO ( 376,380,384,388 ), II
 376  K = 0
      DO 378 J = 1,N
      DO 378 I = 1,N
      K = K + 1
 378  C(K) = VV1(I,J)
      GO TO 392
 380  K = 0
      DO 382 J = 1,N
      DO 382 I = 1,N
      K = K + 1
 382  C(K) = VV2(I,J)
      GO TO 392
 384  K = 0
      DO 386 J = 1,N
      DO 386 I = 1,N
      K = K + 1
 386  C(K) = VV3(I,J)
      GO TO 392
 388  K = 0
      DO 390 J = 1,N
      DO 390 I = 1,N
      K = K + 1
 390  C(K) = VV4(I,J)
C
C ( Q * V(I) * Q * V(J) ) N*N
C
 392  CALL GMPRD(F,C,E,N,N,N)
      K = 0
      DO 394 J = 1,N
      DO 394 I = 1,N
      K = K + 1
 394  VJVI(I,J) = E(K)
C
C TR( Q * V(J) * Q *, V(I) )
C
      CALL TRACE(VJVI,N,SUM)
      TR(JJ,II) = SUM
 396  CONTINUE
 398  CONTINUE
      K = 0
      DO 400 J = 1,4
      DO 400 I = 1,4
      K = K + 1
      C(K) = TR(I,J)
 400  F(K) = TR(I,J)
      CALL MFGR(F,4,4,EPS,IRANK,IROW,ICOL)
      IF ( IRANK - 4 ) 99,402,99
 402  CALL MINV(C,4,D,IROW,ICOL )
      CALL GMPRD(C,S,E,4,4,1)
      DO 403 J = 1,4
      IF ( E(J) .LT. 0 ) E(J) = 0
 403  CONTINUE
```

```
      IF ( ND .NE. 0 ) GO TO 404
C
C  VARIANCE COMPONENT BY MIVQUE METHOD
C
      DO 401 J = 1,4
 401  VE(3,J) = E(J)
C
C  VARIANCE COMPONENT BY I-MIVQUE METHOD
C
 404  DO 405 J = 1,4
 405  VE(4,J) = E(J)
      DO 406 J = 1,4
      IF ( ABS(VE(4,J) - VR(J) ) .GT. 0.05 ) GO TO 418
 406  CONTINUE
      KOUNT1 = KOUNT1 + 1
      WRITE(6,333) KOUNT1,((VE(I,J),J=1,4),I=1,4)
 333  FORMAT(T1,I4,16(F7.2,1X) )
      GO TO 430
 418  ND = ND + 1
      IF ( ND .GE. 20 ) GO TO 99
      DO 420 J = 1,4
 420  VR(J) = VE(4,J)
      GO TO 302
C
C  CUMULATIVE STATISTICS
C
 430  DO 432 I = 1,4
      DO 432 J = 1,4
      SS(I,J) = SS(I,J) + ( (VE(I,J)-SIGMAS(J) ) * * 2)
 432  CONTINUE
      IF ( KOUNT1 .GE. 200 ) GO TO 434
      GO TO 99
 434  DO 436 I = 1,4
      DO 436 J = 1,4
      MSE(I,J) = SS(I,J) / KOUNT1
 436  SMSE(I) = SMSE(I) + MSE(I,J)
      DO 438 I = 1,4
 438  WRITE(6,440) I,(MSE(I,J),J=1,4),SMSE(I)
 440  FORMAT(T2,'MSE OF METHOD ',I1,' = ',5(F20.4,1X) )
      DO 442 I = 1,4
 442  WRITE(6,446) (SS(I,J),J=1,4)
 446  FORMAT(T1,4(F20.5,1X))
      STOP
      END
```

```
C
C.....
C.....                                                              .....
C.....                        SUBROUTINE MINV                       .....
C.....                        ===============                       .....
C.....                                                              .....
C.....            FIND INVERSE MATRIX OF A AND GIVING IN A          .....
C.....               DETERMINANT OF A GIVING IN D                   .....
C.....                                                              .....
C.....
C
      SUBROUTINE MINV(A,N,D,L,M)
      DIMENSION A(1),L(1),M(1)
C     DOUBLE PRECISION A,D,BIGA,HOLD
C
      D = 1.0
      NK = -N
      DO 80 K=1,N
      NK = NK + N
      L(K) = K
      M(K) = K
      KK = NK + K
      BIGA = A(KK)
      DO 20 J = K,N
      IZ = N * (J-1)
      DO 20 I = K,N
      IJ = IZ + I
10    IF( ABS(BIGA)- ABS(A(IJ))) 15,20,20
15    BIGA = A(IJ)
      L(K) = I
      M(K) = J
20    CONTINUE
C
C   INTERCHANGE ROWS
C
      J = L(K)
      IF (J-K) 35,35,25
25    KI = K-N
      DO 30 I =1,N
      KI = KI + N
      HOLD = -A(KI)
      JI = KI - K + J
      A(KI) = A(JI)
30    A(JI) = HOLD
C
C    INTERCHANGE COLUMNS
C
35    I = M(K)
      IF ( I-K) 45,45,38
38    JP = N * (I-1)
      DO 40 J =1,N
      JK = NK + J
      JI = JP + J
      HOLD = - A(JK)
      A(JK) = A(JI)
40    A(JI) = HOLD
C
C   DIVIDED COLUMN BY MINUS PIVOT(VALUE OF PIVOT SYSTEM IS CONTAINED IN
C    BIGA)
C
45    IF (BIGA) 48,46,48
46    D = 0.0
      RETURN
48    DO 55 I=1,N
      IF (I-K) 50,55,50
50    IK = NK + I
      PPP = A(IK)  / (-1 * BIGA)
      A(IK) = PPP
55    CONTINUE
```

```
C     RECODE MATRIX
C
      DO 65 I=1,N
      IK = NK + I
      HOLD = A(IK)
      IJ = I - N
      DO 65 J =1,N
      IJ = IJ + N
      IF (I-K) 60,65,60
60    IF (J-K) 62,65,62
62    KJ = IJ -I + K
      A(IJ) = HOLD * A(KJ) +A(IJ)
65    CONTINUE
C
C     DIVIDED ROW BY PIVOT
C
      KJ = K - N
      DO 75 J=1,N
      KJ =  KJ + N
      IF(J-K) 70,75,70
70    A(KJ) = A(KJ) / BIGA
75    CONTINUE
C
C     PRODUCT OF PIVOT
C
      D = D * BIGA
C
C     REPLACE PIVOT BY RECIPROCAL
C
      A(KK) = 1.0 / BIGA
80    CONTINUE
C
C    FINAL ROW AND COLUMN INTERCHANGE
C
      K = N
100   K = (K-1)
      IF (K) 150,150,105
105   I = L(K)
      IF (I-K) 120,120,108
108   JQ = N * (K-1)
      JR = N * (I-1)
      DO 110 J=1,N
      JK = JQ + J
      HOLD = A(JK)
      JI = JR + J
      A(JK) = -A(JI)
110   A(JI) = HOLD
120   J = M(K)
      IF(J-K) 100,100,125
125   KI = K - N
      DO 130 I=1,N
      KI =KI + N
      HOLD = A(KI)
      JI = KI - K + J
      A(KI) = - A(JI)
130   A(JI) = HOLD
      GO TO 100
150   RETURN
      END
```

```
C
C.....
C.....                                                              .....
C.....                      SUBROUTINE MFGR                         .....
C.....                      ===============                         .....
C.....                                                              .....
C.....                  FIND RANK OF MATRIX A                       .....
C.....                                                              .....
C.....
C
      SUBROUTINE MFGR(A,M,N,EPS,IRANK,IROW,ICOL)
      DIMENSION A(1),IROW(1),ICOL(1)
C     DOUBLE PRECISION A,PIV,HOLD,SAVE
      MN = N * M
      DO 100 J = 1,MN
100   CONTINUE
      IF(M) 2,2,1
1     IF(N) 2,2,4
2     IRANK = -1
3     RETURN
C
C   INITIALIZE COLUMN INDEX VECTOR
C   SEARCH FIRST PIVOT ELEMENT
C
4     IRANK = 0
      PIV = 0.
      JJ =0
      DO 6 J=1,N
      ICOL(J) = J
      DO 6 I=1,M
      JJ = JJ + 1
      HOLD = A(JJ)
      IF( ABS(PIV) - ABS(HOLD)) 5,6,6
5     PIV = HOLD
      IR = I
      IC = J
6     CONTINUE
C
C   INITIALIZE ROW INDEX VECTOR
C
      DO 7 I =1,M
7     IROW(I) = I
C
C   SET UP INTERNAL TOERANCE
C
      TOL = ABS(EPS * PIV)
C
C   INITIALIZE ELEMENT LOOP
C
      NM = N * M
      DO 49 NCOL = M,NM,M
C
C   TEST FOR FEASIBILITY OF PIVOT ELEMENT
C
8     IF( ABS(PIV) - TOL) 20,20,9
C
C   UPDATE RANK
C
9     IRANK = IRANK + 1
C
C   INTERCHANGE ROWS IF NECESSARY
C
      JJ = IR - IRANK
      IF(JJ ) 12,12,10
10    DO 11 J=IRANK,NM,M
      I = J + JJ
      SAVE = A(J)
      A(J) = A(I)
11    A(I) = SAVE
C
C   UPDATE ROW INDEX VECTOR
C
      JJ = IROW(IR)
      IROW(IR) = IROW(IRANK)
      IROW(IRANK) = JJ
C
C   INTERCHANGE COLUMN IF NECESSARY
C
12    JJ = (IC - IRANK) * M
      IF(JJ) 15,15,13
13    KK = NCOL
```

```
      DO 14 J=1,M
      I = KK + JJ
      SAVE = A(KK)
      A(KK) = A(I)
      KK = KK - 1
14    A(I) = SAVE
C
C   UPDATE COLUMN INDEX VECTOR
C
      JJ = ICOL(IC)
      ICOL(IC) = ICOL(IRANK)
      ICOL(IRANK) = JJ
15    KK = IRANK + 1
      MM = IRANK - M
      LL = NCOL + MM
C
C   TEST FOR LAST ROW
C
      IF (MM) 16,25,25
C
C   TRANSFORM CURRENT SUBMATRIX AND SEARCH NEXT PIVOT
C
16    JJ = LL
      SAVE = PIV
      PIV =0.
      DO 39 J=KK,M
      JJ = JJ + 1
      HOLD = A(JJ) / SAVE
      A(JJ) = HOLD
      L = J - IRANK
C
C   TEST FOR LAST COLUMN
C
      IF(IRANK - N) 17,39,39
17    II = JJ
      DO 19 I=KK,N
      II = II + M
      MM = II - L
      A(II) = A(II) - HOLD * A(MM)
      IF ( ABS(A(II)) - ABS(PIV) ) 19,19,18
18    PIV = A(II)
      IR = J
      IC = I
19    CONTINUE
39    CONTINUE
49    CONTINUE
C
C   SET UP MATRIX EXPRESSION ROW DEPENDENCIES
C
20    IF ( IRANK - 1 ) 3,25,21
21    IR = LL
      DO 24 J = 2,IRANK
      II = J - 1
      IR = IR - M
      JJ = LL
      DO 23 I = KK,M
      HOLD = 0.
      JJ = JJ + 1
      MM = JJ
      IC = IR
      DO 22 L = 1,II
      HOLD = HOLD + A(MM) * A(IC)
      IC = IC - 1
22    MM = MM - M
23    A(MM) = A(MM) - HOLD
24    CONTINUE
```

```
C
C   TEST FOR COLUMN REGULARITY
C
25    IF ( N - IRANK ) 3,3,26
C
C   SET UP MATRIX EXPRESSING BASIC VARIABLES IN TERM OF FREE
C   PARAMETERS ( HOMOGENEOUS SOLUTION )
C
26    IR = LL
      KK = LL + M
      DO 30 J = 1,IRANK
      DO 29 I = KK,NM,M
      JJ = IR
      LL = I
      HOLD = 0.
      II = J
27    II = II - 1
      IF ( II ) 29,29,28
28    HOLD = HOLD - A(JJ) * A(LL)
      JJ = JJ - M
      LL = LL - 1
      GO TO 27
29    A(LL) = ( HOLD- A(LL) ) / A(JJ)
30    IR = IR - 1
      RETURN
      END
```

```
C
C.....
C.....                                                          .....
C.....                        SUBROUTINE GMPRD                  .....
C.....                        ================                  .....
C.....                                                          .....
C.....                                                          .....
C.....          FOR MULTIPLY MATRIX A AND B GIVING MATRIX R     .....
C.....                                                          .....
C.....
C
      SUBROUTINE GMPRD( A,B,R,N,M,L )
      DIMENSION A(1),B(1),R(1)
C     DOUBLE PRECISION A,B,R
C
      IR = 0
      IK = - M
      DO 10 K = 1,L
      IK = IK + M
      DO 10 J = 1,N
      IR = IR + 1
      JI = J - N
      IB = IK
      R(IR) = 0
      DO 10 I = 1,M
      JI = JI + N
      IB = IB + 1
10    R(IR) = R(IR) + A(JI) * B(IB)
      RETURN
      END
```

```
C
C.....
C.....                                                          .....
C.....                      SUBROUTINE TRACE                    .....
C.....                      ================                    .....
C.....                                                          .....
C.....                  FIND TRACE OF MATRIX A                  .....
C.....                                                          .....
C.....
C
      SUBROUTINE TRACE(A,N,SUM)
      DIMENSION A(30,30)
C     DOUBLE PRECISION A,SUM
      SUM = 0.0
      DO 10 I = 1,N
      DO 10 J = 1,N
      IF ( I - J ) 10,20,10
20    SUM = SUM + A(I,J)
10    CONTINUE
      RETURN
      END
```

```
C
C.....
C.....                                                              .....
C.....                        FUNCTION  NORM                        .....
C.....                        =============                         .....
C.....                                                              .....
C.....          FOR GENERATE DATA WITH NORMAL DISTRIBUTION          .....
C.....                                                              .....
C.....
C
      FUNCTION NORM( ISEED,SIGMA,MEAN )
      REAL NORM
      DATA PI/3.14159/,L1/0/
C
C   CHECK TO SEE WHICH N(0,1) RANDOM VARIATE TO USE
C
C
      IF ( L1 .EQ. 1 ) GO TO 10
C
C   GENERATE TWO UNIFORM RANDOM NUMBERS
C
      CALL RANDU( ISEED,IYY,YYFL )
      RONE = YYFL
      ISEED = IYY
C
      CALL RANDU(ISEED,IYY,YYFL)
      RTWO = YYFL
      ISEED = IYY
C
C  GENERATE TWO NORMAL(0,1) RANDOM VARIATBLES
C
      ZONE = SQRT( -2 * ALOG( RONE ) ) * COS( 2 * PI * RTWO )
      ZTWO = SQRT( -2 * ALOG( RONE ) ) * SIN( 2 * PI * RTWO )
C
C  COMPUTE NORMAL RANDOM VARIABLE WITH PARAMETER ( MEAN,SIGMA )
C  FOR MEAN AND STANDARD DEVIATION
C
      NORM = ZONE * SIGMA + MEAN
      L1 = 1
      RETURN
C
C   COMPUTE NORMAL RANDOM VARIABLE , N( MEAN , SIGMA ** 2 )
C
 10   NORM = ZTWO * SIGMA + MEAN
      L1 = 0
      RETURN
      END
```

```
C
C.....
C.....                                                              .....
C.....                        SUBROUTINE RANDU                      .....
C.....                        ================                      .....
C.....                                                              .....
C.....   FOR GENERATE RANDOM NUMBER WITH UNIFORM DISTRIBUTION       .....
C.....                                                              .....
C.....
C
      SUBROUTINE RANDU( IX,IY,YFL )
C
      IY = IX * 65539
      IF (IY) 5,6,6
    5 IY = IY + 2147483647 + 1
    6 YFL = IY
      YFL = YFL / 2147483647
      RETURN
      END
```

ประวัติผู้เขียน

นางสาวยุพิน คำเหมือง เกิดเมื่อวันที่ 14 มิถุนายน พ.ศ. 2503 ที่จังหวัดอุตรดิตถ์ สำเร็จการศึกษาปริญญาตรี วิทยาศาสตรบัณฑิต (สถิติ) จากมหาวิทยาลัยเชียงใหม่ เมื่อปีการศึกษา 2525 และเข้าศึกษาต่อระดับปริญญาโทในภาควิชาสถิติ จุฬาลงกรณ์มหาวิทยาลัย ปีการศึกษา 2526 ปัจจุบันทำงานอยู่ที่กองการสอบ สำนักงาน ก.พ.