COMPLETION TECHNICAL REPORT
06

EVALUATION OF PARAMETER ESTIMATION METHODS FOR FLOOD FREQUENCY ANALYSIS: COMPUTER PROGRAMS

by

VIJAY P. SINGH
KISHORE ARORA

for

U. S. GEOLOGICAL SURVEY
Department of Interior
Reston, VA 22092

LOUISIANA WATER RESOURCES RESEARCH INSTITUTE
Louisiana State University
Baton Rouge, LA 70803

EVALUATION OF PARAMETER ESTIMATION METHODS FOR FLOOD FREQUENCY ANALYSIS: COMPUTER PROGRAMS

Kishore Arora and Vijay P. Singh
Department of Civil Engineering
Louisiana State University
Baton Rouge, LA 70803-6405

March 1987

ACKNOWLEDGMENTS

This study was supported in part by the U.S. Department of Interior, Geological Survey, through the Louisiana Water Resources Research Institute, under the project, "A Multivariate Stochastic Analysis of Flood Magnitude, Duration and Volume." The computer programs presented here were utilized to complete an M.S. thesis of the senior author. Helpful suggestions of Drs. J. F. Cruise, H. Schneider and M. Fiorentino are gratefully acknowledged.

COMPUTER PROGRAMS

These programs were used to assess the statistical performance indices of various estimators available for the flood frequency distribution under consideration. This involved three main phases of development: (1) generation of the pseudo-random numbers for the respective distribution, (2) estimation of parameters and quantiles by each of the available methods, and (3) computation of performance indices of bias, standard error, and root mean square error of the estimator.

The generation of the random numbers for the extreme value type I (EV1) and the two-component extreme value distributions is straightforward. It was accomplished by using the inverse form of the EV1 distribution, and generating the uniform and standard normal random numbers through the IBM supported IMSL generators. Several alternative generation schemes were tested for generating the log Pearson type 3 (LP3) numbers. The scheme found most suitable (SUBROUTINE LP3GN4) consisted of generating one parameter gamma variates using IMSL routine and transforming them to LP3 numbers.

Each of the available estimation methods for a given distribution was programmed as independent subroutine(s). These routines were thoroughly checked for round-off and convergence problems and debugged before using them in final simulation runs.

Finally, after obtaining the parameter and quantile estimates from each of the estimators, their performance indices were calculated in a straightforward manner.

The user can easily combine the various subroutines to obtain the parameter and quantile estimates by various estimators (estimation

method plus distribution) in a single computer run. A generous use is made of various standard IMSL routines to perform a variety of lower level calculations. These routine calls may have to be suitably replaced depending upon availability in a particular environment. All programs have been run on WATFIV compilers supported by MVO/TSO or IBM 370/3033-3084.

TABLE OF CONTENTS

```
      DOUBLE PRECISION DSEED
      REAL MR,MR2,MR3
      BINV=1.0/(9.0*B)                                          2
      SR=0.0
      SR2=0.0
      SR3=0.0
C
      CALL GGNML(DSEED,NR,R)
C
      DO 10 I=1,NR
      FAC=1.0-BINV+R(I)*(SQRT(BINV))
      FAC=(FAC*FAC*FAC*A*B)+C
      R(I)=EXP(FAC)
C
      SR=SR+R(I)
      SR2=SR2+R(I)*R(I)
      SR3=SR3+R(I)*R(I)*R(I)
C
   10 CONTINUE
      WRITE(6,*)SR,SR2,SR3
      XNR=FLOAT(NR)
      MR=SR/XNR
      MR2=SR2/XNR
      MR3=SR3/XNR
      WRITE(6,*)MR,MR2,MR3
C
      RAVG=MR
      RVAR=(MR2-MR*MR)*(XNR/(XNR-1.0))
      RSKW=(MR3+2*(RAVG**3.0)-3*RAVG*MR2)/(RVAR**1.5)
      RSKW=RSKW*(XNR/(XNR-1.0))*(XNR/(XNR-2.0))
      RETURN
      END
C
      SUBROUTINE LP3GN2(A,B,C,DSEED,NR)
      COMMON/LP3NUM/R(100100)
      COMMON/STAT/RAVG,RVAR,RSKW
      DOUBLE PRECISION DSEED
      DOUBLE PRECISION DT,ST,DR,DA,DC
      REAL MR,MR2,MR3
      DA=A
      DC=C
      IB=IFIX(B)
      ST=0.0D0
      SR=0.0
      SR2=0.0
      SR3=0.0
      DO 10 I=1,NR
      DO 20 J=1,IB
      T=GGUBFS(DSEED)
      DT=T
      ST=ST+DLOG(DT)
   20 CONTINUE
      DR=DC-DA*ST
      R(I)=DEXP(DR)
C
      SR=SR+R(I)
      SR2=SR2+R(I)*R(I)
      SR3=SR3+R(I)*R(I)*R(I)
C
      ST=0.0D0
```

```
   10  CONTINUE
       WRITE(6,*)SR,SR2,SR3
       XNR=FLOAT(NR)
       MR=SR/XNR
       MR2=SR2/XNR
       MR3=SR3/XNR
       WRITE(6,*)MR,MR2,MR3
C
       RAVG=MR
       RVAR=(MR2-MR*MR)*(XNR/(XNR-1.0))
       RSKW=(MR3+2.*(RAVG**3.0)-3.*RAVG*MR2)/(RVAR**1.5)
       RSKW=RSKW*(XNR/(XNR-1.0))*(XNR/(XNR-2.0))
       RETURN
       END
C
       SUBROUTINE LP3GN3(A,B,C,DSEED,NR)
       COMMON/LP3NUM/R(100100)
       COMMON/STAT/RAVG,RVAR,RSKW
       DOUBLE PRECISION DSEED
       REAL MR,MR2,MR3
C
       BRT=SQRT(B)
       SK=2.0/BRT
C
       T=2.0/SK
       AA=AMAX1(T,0.40)
C
       T=SK-2.25
       BB=AMAX1(0.,T)
       BB=1.+0.0144*BB*BB
C
       T=SK-1.0
       GG=AMAX1(0.,T)
       GG=SK-0.063*(GG**1.85)
C
C      HH=BB-(2.0/(SK*AA))
C      HH=HH**(1./3.)
       HH=0.256113
C
       SR=0.0
       SR2=0.0
       SR3=0.0
C
       CALL GGNML(DSEED,NR,R)
C
       FAC=GG/6.0
       DO 10 I=1,NR
       WH=1.0-(FAC*FAC)+FAC*R(I)
       P=AMAX1(HH,WH)
       WHM=AA*(P*P*P-BB)
       XP=(A*BRT*(WHM+BRT))+C
       R(I)=EXP(XP)
C
       SR=SR+R(I)
       SR2=SR2+R(I)*R(I)
       SR3=SR3+R(I)*R(I)*R(I)
C
   10  CONTINUE
       WRITE(6,*)SR,SR2,SR3
       XNR=FLOAT(NR)
```

```
      MR=SR/XNR
      MR2=SR2/XNR
      MR3=SR3/XNR
      WRITE(6,*)MR,MR2,MR3
C
      RAVG=MR
      RVAR=(MR2-MR*MR)*(XNR/(XNR-1.0))
      RSKW=(MR3+2*(RAVG**3.0)-3*RAVG*MR2)/(RVAR**1.5)
      RSKW=RSKW*(XNR/(XNR-1.0))*(XNR/(XNR-2.0))
      RETURN
      END
C
      SUBROUTINE LP3GN4(A,B,C,DSEED,NR)
      COMMON/LP3NUM/R(100100)
      COMMON/STAT/RAVG,RVAR,RSKW
      DOUBLE PRECISION DSEED
      DOUBLE PRECISION DT,DR,DSR,DSR1,DSR2
      REAL MR,MR2,MR3
      DIMENSION WK(1)
      CALL GGAMR(DSEED,B,NR,WK,R)
      DSR=0.D0
      DSR2=0.D0
      DSR3=0.D0
C
      DO 10 I=1,NR
      T=A*R(I)+C
      DT=T
      DR=DEXP(DT)
      R(I)=DR
C
      DSR=DSR+DR
      DSR2=DSR2+DR*DR
      DSR3=DSR3+DR*DR*DR
C     SR=SR+R(I)
C     SR2=SR2+R(I)*R(I)
C     SR3=SR3+R(I)*R(I)*R(I)
   10 CONTINUE
      SR=DSR
      SR2=DSR2
      SR3=DSR3
C
      WRITE(6,*)' SUMMATION X, X**2, X**3 : ',SR,SR2,SR3
      XNR=FLOAT(NR)
      XNR1=XNR/(XNR-1.)
      XNR2=XNR/(XNR-2.)
      MR=SR/XNR
      MR2=SR2/XNR
      MR3=SR3/XNR
      WRITE(6,*)' 1ST, 2ND, & 3RD MOMENTS : ',MR,MR2,MR3
C
      RAVG=MR
      RVAR=(MR2-MR*MR)*XNR1
      RSKW=(MR3+2*(RAVG**3.0)-3*RAVG*MR2)/(RVAR**1.5)
      RSKW=RSKW*XNR1*XNR2
      WRITE(6,*)' AVERAGE, VARIANCE & SKEW : ',RAVG,RVAR,RSKW
      RETURN
      END
C
$ENTRY
$$
```

```
//
//GO.FT08F001 DD DSN=CEAROR.TCEV.****,DISP=SHR
```

```
//PROJECT JOB (1304,59634,10,20),'ARORA',MSGCLASS=S,CLASS=A
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=N
//     EXEC WATFIV,REGION.GO=4000K,TIME.GO=99
$JOB            TIME=4500
C=================================================================
C                LOG PEARSON TYPE 3 DISTRIBUTION
C-----------------------------------------------------------------
C      COMPUTES PARAMETERS AND QUANTILES BY METHOD OF MOMENT-DIRECT(MMD)
C=================================================================
       COMMON/QUANT/Q10,Q25,Q50,Q100,Q200,Q500
       COMMON/SKEW/SKX
       COMMON/LP3NUM/R(100100),R2(100100),R3(100100)
       COMMON/ALL/BTAB(309),ALPTAB(309)
       COMMON/MMPAR/A,B,C
       COMMON/EST/ALPEST
       COMMON/BVAL/BB
       DIMENSION ISSIZE(10)
       DOUBLE PRECISION DSEED
C=================================================================
C        NCSE = NO. OF CASES OF SAMPLE SIZES TO BE ANALYSED
C         M   = NO. OF MONTE-CARLO SAMPLES FOR EACH NCSE
C         N   = SIZE OF EACH SAMPLE (=ISSIZE(.))
C    AP,BP,CP = POPULATION PARAMETERS OF LP3 DISTRIBUTION
C       NRAN  = TOTAL NO. OF MONTE-CARLO NUMBERS GENERATED FOR GIVEN
C               POPULATION PARAMETERS (M*N < NRAN)
C=================================================================
C
       NCSE=5
       READ(5,*)(ISSIZE(I),I=1,NCSE)
       READ(9,*)(BTAB(I),ALPTAB(I),I=1,309)
C
       M=1000
C
       NRAN=75000
       DSEED=123457.D0
       AP=0.059798
       BP=98.38009
       CP=-6.066213
       COVP=0.7
       SKP= 3.0
C
       WRITE(6,1)
  1    FORMAT(1H1/'  CASE   5   --   C.V. = 0.7,   SKEW = 3 ')
       WRITE(6,*)'                '
       WRITE(6,*)' POPULATION PARM. : ',AP,BP,CP,'| CV, SKEW :',COVP,SKP
       WRITE(10,*)AP,BP,CP,COVP,SKP,DSEED
C-----------------------------------------------------------------
       CALL LP3GN4(AP,BP,CP,DSEED,NRAN)
C
       DO 10 I=1,NCSE
       N=ISSIZE(I)
       WRITE(6,*)' SAMPLE SIZE = ',N,' NO. OF SAMPLES = ',M
       WRITE(10,*)N,M
```

```
C
      DO 20 J=1,M
      KL=N*(J-1)+1
      KU=N*J
      CALL STATS(KL,KU)
      IF(BB.GE.2.0369510)GO TO 25
      WRITE(6,*)' B TOO SMALL : ',BB
      ALPEST=-5000.0
      GO TO 26
   25 CALL POLATE
C     WRITE(6,*)' B = ',BB,' A (INTERP.) = ',ALPEST
   26 CALL MMDIR
C
      XMR=1.
      CPMY=C+B*A
      VARL=B*A*A
      STDL=SQRT(VARL)
      SKL=2.*(ABS(A)/A)/SQRT(B)
      IF(ABS(SKL).LE.5.5)GO TO 30
      WRITE(6,*)' LOG SKEW = ',SKL
      GO TO 40
   30 CALL LPQNTL(CPMY,STDL,SKL,XMR)
C
C     SKXP=SKX/SKP
   40 SKXP=SKP
      WRITE(6,11)J,A,B,C,SKXP,Q10,Q25,Q50,Q100,Q200,Q500
      WRITE(10,11)J,A,B,C,SKXP,Q10,Q25,Q50,Q100,Q200,Q500
   11 FORMAT(1X,I4,F16.7,E15.7,F17.7,F13.7,6F10.5)
   20 CONTINUE
   10 CONTINUE
      STOP
      END
C
C           ********   END OF MAIN SEGMENT   ********
C
C*************************************************************************
C     SUBROUTINE LP3GN4 GENERATES LP3 NUMBERS USING IMSL ROUTINE
C*************************************************************************
      SUBROUTINE LP3GN4(A,B,C,DSEED,NR)
      COMMON/LP3NUM/R(100100),R2(100100),R3(100100)
      DOUBLE PRECISION DSEED
      DOUBLE PRECISION DT,DR
      REAL MR,MR2,MR3
      DIMENSION WK(1)
C
      CALL GGAMR(DSEED,B,NR,WK,R)
C
      DO 10 I=1,NR
      T=A*R(I)+C
      DT=T
      DR=DEXP(DT)
      R(I)=DR
      R2(I)=R(I)*R(I)
      R3(I)=R2(I)*R(I)
   10 CONTINUE
      RETURN
      END
C*************************************************************************
C     SUBROUTINE TO COMPUTE THE FIRST THREE MOMENTS OF THE SAMPLE
C     FROM SAMPLE ESTIMATES OF MEAN,VARIANCE AND SKEW. ALSO
```

```
C     CALCULATES "B" AS A FUNCTION OF THE THREE MOMENTS.
C     B IS USED FOR MMD PARAMETER ESTIMATION
C*************************************************************************
      SUBROUTINE STATS(KL,KU)
      COMMON/LP3NUM/R(100100),R2(100100),R3(100100)
      COMMON/LNMMNT/AL1,AL2,AL3
      COMMON/BVAL/B
      COMMON/SKEW/SKX
      REAL L1,L2,L3
      DOUBLE PRECISION DL1,DL2,DL3,DAL1,DAL2,DAL3,DB
C
      CALL UBVSK(KL,KU,L1,L2,L3)
C
C     L1=XM
C     XM2=XM*XM
C     L2=XM2+VARX
C     L3=XM2*XM+3.*XM*VARX+(VARX**1.5)*SKX
C
      DL1=L1
      DL2=L2
      DL3=L3
      DAL1=DLOG(DL1)
      DAL2=DLOG(DL2)
      DAL3=DLOG(DL3)
      AL1=DAL1
      AL2=DAL2
      AL3=DAL3
      DB=(DAL3-3.0*DAL1)/(DAL2-2.0*DAL1)
      B=DB
      RETURN
      END
C*************************************************************************
C    UBVSK: SUB-ROUTINE TO COMPUTE THE FIRST THREE ORIGIN MOMENTS OF
C    THE REAL (UNTRANSFORMED) DATA
C*************************************************************************
      SUBROUTINE UBVSK(KL,KU,VM1,VM2,VM3)
      COMMON/LP3NUM/V(100100),V2(100100),V3(100100)
      FN=FLOAT(KU-KL+1)
C     C1=FN/(FN-1.)
C     C2=FN**2/(FN-1.)/(FN-2.)
C     C2=C2/C1**1.5
      X1=0.
      X2=0.
      X3=0.
      DO 10 I=KL,KU
      X1=X1+V(I)
      X2=X2+V2(I)
 10   X3=X3+V3(I)
      VM1=X1/FN
      VM2=X2/FN
      VM3=X3/FN
C     VM=X1/FN
C     VAV=X2/FN-VM**2
C     SKV=(X3/FN-3.*VM*VAV-VM**3)/VAV**1.5
C     VAV=VAV*C1
C     SKV=SKV*C2
C
C     CORR=1.+8.5/FN
C     SKV=SKV*CORR
      RETURN
```

```
      END
C
C*********************************************************************** 9
C     SUBROUTINE POLATE INTERPOLATES FOR DIRECT METHOD OF MOMENTS
C     USING BTAB(.),ALPTAB(.)
C***********************************************************************
      SUBROUTINE POLATE
      COMMON/BVAL/B
      COMMON/EST/ALPEST
      COMMON/ALL/BTAB(309),ALPTAB(309)
      DO 10 I= 1,309
      IF((B .LT. 2.036951).OR. (B .GT.8.561942))GO TO 12
      IF((B.GE.BTAB(I)).AND. (B .LE. BTAB(I+1)))GO TO 50
 10   CONTINUE
 50   DELALP= (ALPTAB(I)-ALPTAB(I+1))/(BTAB(I)-BTAB(I+1))
     1*(B-BTAB(I+1))
      ALPEST= DELALP+ALPTAB(I+1)
      RETURN
 12   WRITE(6,*)' B = ',B,' IS OUT OF RANGE : (2.036951 - 8.561942) '
      RETURN
      END
C
C*********************************************************************
C     SUBROUTINE TO CALCULATE THE PARAMETERS BY METHOD OF MOMENTS
C*********************************************************************
      SUBROUTINE MMDIR
      COMMON/EST/ALPEST
      COMMON/MMPAR/ALPHA,BETA,GAMMA
      COMMON/LNMMNT/AL1,AL2,AL3
      DOUBLE PRECISION DAL1,DAL2,DALPHA,DA1,DA2
      DAL1=AL1
      DAL2=AL2
C
      CALL ROOT(ALPEST)
C
      ALPHA=ALPEST
      DALPHA=ALPEST
      DA1=DLOG(1.D0-DALPHA)
      DA2=DLOG(1.D0-2.D0*DALPHA)
      BETA=(DAL2-2.D0*DAL1)/(2.D0*DA1-DA2)
      A1=DA1
      GAMMA=AL1+BETA*A1
      RETURN
      END
C
C=====================================================================
C     ROOT : REFINES THE INTERPOLATED ROOT OF PARAMETER 'A' BY NEWTON
C            RAPHSON METHOD...CALLED IN ROUTINE MMDIR
C=====================================================================
      SUBROUTINE ROOT(A)
      COMMON/BVAL/B1
      DOUBLE PRECISION DA,DAA,DA1,DA2,DA3,DAL1,DAL2,DAL3,TN,TD,B
      DA=A
      B=B1
  10  DA1=1.D0-DA
      DA2=1.D0-2.D0*DA
      DA3=1.D0-3.D0*DA
      DAL1=DLOG(DA1)
      DAL2=DLOG(DA2)
      DAL3=DLOG(DA3)
```

```
      TN=3.D0*DAL1-DAL3-2.D0*B*DAL1+B*DAL2
      TD=(2.D0*B-3.D0)/DA1-2.D0*B/DA2+3.D0/DA3
C     IF(DABS(TN).LT.1.0D-10)GO TO 20                                    10
C     WRITE(6,*)TN,TD,DA
      DAA=DA-TN/TD
      IF(DABS(DAA-DA).LT.1.0D-06)GO TO 20
      DA=DAA
      GO TO 10
 20   A=DAA
      RETURN
      END
C
C*************************************************************************
C--LPQNTL - SUB-ROUTINE TO COMPUTE LOG PEARSON QUANTILES. PEARSON
C  FACTORS GIVEN IN WRC BULLETIN #17 (K-TABLES) ARE LINEARLY
C  INTERPOLATED
C*************************************************************************
C
      SUBROUTINE LPQNTL(XM,STD,SK,XMR)
      COMMON/QUANT/Q10,Q25,Q50,Q100,Q200,Q500
      DIMENSION XK(15,111),CDF(15),RTPK(15),RTLF(15),K(15),Q(15),
     1XJ(111),X1(111),X2(111),X3(111),X4(111),X5(111),X6(111),
     1X7(111),X8(111),X9(111),X10(111),X11(111),X12(111),
     1X13(111),X14(111),X15(111)
      REAL K
      DATA CDF/.005,.01,.02,.04,.1,.2,.5,.8,.9,.96,.98,.99,.995,.998,
     1.999/
      DATA X1/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.606,-.625,-.6452,
     3-.6667,-.6896,-.7143,-.7407,-.7691,-.7997,-.8328,-.8686,-.9074,
     4-.9495,-.995,-1.0443,-1.0975,-1.1548,-1.2162,-1.2817,-1.3511,
     5-1.4244,-1.5011,-1.5811,-1.6639,-1.7492,-1.8366,-1.9258,-2.0164,
     6-2.10825,-2.2009,-2.2942,-2.388,-2.4819,-2.5758,
     70.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,
     80.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,25*0./
      DATA X2/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.6061,-.625,-.6451,
     1-0.6666,-0.6896,-.7145,-0.7405,-0.7688,-0.7992,-0.832,-0.8672,
     1-0.9052,-0.9461,-0.99,-1.037,-1.0871,-1.1404,-1.1968,-1.2561,
     2-1.3182,-1.3827,-1.4494,-1.5181,-1.5884,-1.66,-1.7327,-1.8062,
     3-1.8803,-1.9547,-2.0293,-2.1039,-2.1784,-2.2526,-2.3264,55*0./
      DATA X3/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.6061,-.625,-.6451,
     1-0.6665,-0.6894,-.7138,-0.7399,-0.7678,-0.79765,-0.8296,-0.8637,
     1-0.9001,-0.9388,-0.9798,-1.0231,-1.0686,-1.1163,-1.1658,-1.2172,
     1-1.27,-1.3241,-1.3793,-1.4353,-1.4919,-1.5489,-1.606,-1.6633,
     2-1.7203,-1.7772,-1.8336,-1.8896,-1.945,-1.9997,-2.0538,55*0./
      DATA X5/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44443,-.45452,-.4651,-.4761,-.4877,-.4999,
     2-.5126,-.526,-.5401,-.5548,-.57035,-.5867,-.6038,-.62175,-.6406,
     1-0.6602,-0.6808,-.7021,-0.7242,-0.7471,-0.7706,-0.7947,
     1-0.8193,-0.8442,-0.8694,-0.8946,-0.9199,-0.945,-0.9698,-0.9942,
     2-1.0181,-1.0414,-1.0641,-1.0861,-1.1073,-1.1276,-1.1471,-1.1657,
     3-1.1835,-1.2003,-1.2162,-1.2311,-1.2452,-1.2582,-1.2704,-1.2816,
     455*0./
      DATA X6/-.3636,-.3704,-.37734,-.38458,-.39211,-.39993,-.40806,
     1-.4165,-.4253,-.4345,-.444,-.454,-.4643,-.475,-.4862,-.4978,
```

```
2-.5099,-.5224,-.5353,-.5487,-.5624,-.5765,-.591,-.6057,-.6206,
1-0.6357,-0.6509,-.666,-0.6811,-0.696,-0.7107,-0.725,-0.7388,
1-0.7521,-0.7648,-0.7769,-0.7882,-0.7987,-0.8084,-0.8172,-0.8252,
2-.8322,-.8384,-.8437,-.8481,-.8516,-.8543,-.8561,-.857,-.8572,
3-.8565,-.8551,-.8529,-.8499,-.8461,-.8416,55*0./
 DATA X7/-.3546,-.3596,-.3645,-.3695,-.3743,-.379,-.3836,-.388,
1-.3922,-.3962,-.3999,-.4032,-.4062,-.4088,-.411,-.4127,-.4138,
2-.4144,-.4144,-.4138,-.4125,-.4106,-.4079,-.4045,-.4004,
1-.3955,-.3899,-.3835,-.3764,-.3685,-.3599,-.3506,-.3406,
1-.33,-.3187,-.3069,-.2944,-.2815,-.2681,-.2542,-.24,-.2254,
2-.2104,-.1952,-.1797,-.164,-.1481,-.132,-.1158,-.0995,-.083,
3-.0665,-.0499,-.0333,-.0166,0.,55*0./
 DATA X8/-.0103,.00243,.0156,.0293,.0434,.058,.073,.0885,.1044,
1.1207,.1374,.1545,.1719,.1897,.2078,.2262,.2448,.2638,.2829,.3022,
2.3217,.3413,.361,.3808,.4006,
1.4204,.4402,.4598,.4793,.4987,.5179,.5368,.5555,.5738,
1.5918,.6094,.6266,.6434,.6596,.6753,.6905,.7051,.7192,.7326,
2.7454,.7575,.769,.7799,.79,.7995,.8083,.8164,.8238,.8304,.8364,
3.8416,55*0./
 DATA X9/.6912,.712,.7328,.7536,.7746,.7955,.8164,.8373,.8582,
1.879,.8996,.9202,.9406,.9609,.981,1.0008,1.0204,1.0397,1.0586,
21.0773,1.0955,1.1134,1.1308,1.1477,1.1642,1.1801,1.1954,
11.2101,1.2242,1.2377,1.2504,1.2624,1.2737,1.2841,1.2938,
11.3026,1.3105,1.3176,1.3238,1.329,1.3333,1.3367,1.339,1.3405,
21.3409,1.3404,1.3389,1.3364,1.3329,1.3285,1.3231,1.3167,1.3094,
31.3011,1.2918,1.2816,55*0./
 DATA X10/2.0474,2.0637,2.0795,2.0949,2.1099,2.1243,2.1383,2.1517,
12.1647,2.177,2.1887,2.1999,2.2104,2.2202,2.2294,2.2379,2.2456,
22.2525,2.2587,2.2641,2.2686,2.2723,2.2751,2.2769,2.2779,2.2778,
12.27676,2.2747,2.2716,2.2674,2.2622,2.2558,2.2483,2.2397,
12.2299,2.2189,2.2067,2.1933,2.1787,2.1629,2.1459,2.1277,2.1082,
22.0876,2.0657,2.0427,2.0185,1.9931,1.9666,1.939,1.9102,1.8804,
31.8495,1.8176,1.7846,1.7507,1.7158,1.68,1.6433,1.6057,1.5674,
41.5283,1.4885,1.4481,1.4072,1.3658,1.3241,1.2823,1.2403,1.1984,
51.1568,1.1157,1.0751,1.0354,.9967,.9592,.923,.8881,.8549,.8232,
6.7931,.7646,.7377,.7123,.6884,.6659,
6.6447,.6247,.6059,.5881,.5714,.5555,.5405,.5263,.5128,.5,.4878,
7.4762,.4651,.4546,.4444,.4348,.4255,.4167,.4082,.4,.3922,.3846,
8.3774,.3704,.3636/
 DATA X11/3.2838,3.2884,3.2924,3.2957,3.2982,3.30007,3.3012,
13.3015,3.301,3.2998,3.2977,3.2947,3.2909,3.2862,3.2806,3.274,
23.2665,3.258,3.2485,3.238,3.2264,3.2138,3.2,3.1851,3.1691,
33.1519,3.1336,3.114,3.0932,3.0712,3.0479,3.0233,2.9974,2.9703,
42.9418,2.912,2.8809,2.8485,2.8147,2.7796,2.7433,2.7056,2.6666,
52.6263,2.5848,2.5421,2.4981,2.453,2.4067,2.3593,2.3108,2.2613,
62.2108,2.1594,2.107,2.0538,55*0./
 DATA X12/4.6402,4.6285,4.6159,4.6025,4.5882,4.573,4.5569,4.5399,
14.5219,4.503,4.483,4.4621,4.4401,4.4171,4.393,4.3678,4.3415,4.314,
24.2855,4.2557,4.2247,4.1926,4.1592,4.1245,4.0886,
34.0514,4.0129,3.973,3.9318,3.8893,3.8454,3.8001,3.7535,3.7054,
43.656,3.6052,3.553,3.4994,3.4444,3.388,3.3304,3.2713,3.211,
53.1494,3.0866,3.0226,2.9574,2.891,2.8236,2.7551,2.6857,2.6154,
62.5442,2.4723,2.3996,2.3264,55*0./
 DATA X13/6.08307,6.0517,6.0193,5.986,5.9517,5.9164,5.88,5.8427,
15.8042,5.7646,5.724,5.6822,5.6393,5.5953,5.5501,5.5036,5.456,
25.4071,5.357,5.3056,5.2529,5.1989,5.1436,5.087,5.029,
34.9696,4.9088,4.8467,4.7831,4.7182,4.6518,4.5839,4.5147,4.444,
44.3719,4.2983,4.2234,4.147,4.0693,3.9902,3.9097,3.828,3.745,
53.6607,3.5753,3.4887,3.4011,3.3124,3.2228,3.1323,3.041,2.949,
62.8564,2.7632,2.6697,2.5758,55*0./
```

```
      DATA X14/8.0869,8.0259,7.9639,7.9008,7.8366,7.7712,7.7048,
     17.6372,7.5684,7.4985,7.4273,7.355,7.2814,7.2065,7.1304,7.053,
     26.9744,6.8944,6.813,6.7303,6.6463,6.5608,6.474,6.3858,6.2961,
     16.20506,6.1125,6.0186,5.9232,5.8263,5.728,5.6282,5.5269,
     15.4243,5.3201,5.2146,5.1077,4.9994,4.8897,4.7788,4.6665,4.553,
     24.4384,4.3226,4.2058,4.088,3.9693,3.8498,3.7296,3.6087,3.4874,
     33.3657,3.2437,3.1217,2.9998,2.8782,2.7571,2.6367,2.5174,
     42.3994,2.2831,2.1688,2.057,1.9481,1.8424,1.7406,1.6431,1.5502,
     51.4623,1.3798,1.3028,1.2313,1.1653,1.1047,1.049,.998,.9513,.9085,
     6.8693,.8332,.7999,.7692,.7407,.7143,.6896,.6667,25*0./
      DATA X15/9.6577,9.5723,9.4859,9.3983,9.3095,9.2196,9.1285,9.0362,
     18.9427,8.848,8.752,8.6548,8.5563,8.4565,8.3553,8.2529,8.1491,
     28.044,7.9374,7.8295,7.7202,7.6095,7.4974,7.3838,7.2688,
     17.1524,7.0344,6.9151,6.7942,6.6719,6.5481,6.4229,6.2963,
     26.1682,6.0387,5.9078,5.7755,5.6419,5.507,5.3709,5.2335,5.0951,
     34.9555,4.8149,4.6734,4.5311,4.3881,4.2444,4.1002,3.9557,3.8109,
     43.6661,3.5214,3.377,3.2332,3.0902,2.9483,2.8079,2.6692,2.5326,
     52.3987,2.2678,2.1405,2.0174,1.8989,1.7857,1.6783,1.577,1.4822,
     61.3941,1.3128,1.2381,1.1697,1.1074,1.0507,.999,.9519,.9089,.8695,
     7.8333,.8,.7692,.7407,.7143,.6897,.6667,25*0./
      J=111
      DO 61 I=1,25
      X14(J)=-X1(I)
      X15(J)=-X1(I)
61    J=J-1
      J=111
      DO 62 I=1,111
      X4(J)=-X10(I)
62    J=J-1
      J=111
      DO 63 I=1,55
      X1(J)=-X13(I)
      X2(J)=-X12(I)
      X3(J)=-X11(I)
      X5(J)=-X9(I)
      X6(J)=-X8(I)
      X7(J)=-X7(I)
      X8(J)=-X6(I)
      X9(J)=-X5(I)
      X11(J)=-X3(I)
      X12(J)=-X2(I)
      X13(J)=-X1(I)
63    J=J-1
      DO 1 J =1,111
      XK(1,J)=X1(J)
      XK(2,J)=X2(J)
      XK(3,J)=X3(J)
      XK(4,J)=X4(J)
      XK(5,J)=X5(J)
      XK(6,J)=X6(J)
      XK(7,J)=X7(J)
      XK(8,J)=X8(J)
      XK(9,J)=X9(J)
      XK(10,J)=X10(J)
      XK(11,J)=X11(J)
      XK(12,J)=X12(J)
      XK(13,J)=X13(J)
      XK(14,J)=X14(J)
      XK(15,J)=X15(J)
 1    CONTINUE
```

```
      DO 65 I=1,15
      RTLF(I)=1./CDF(I)
 65   RTPK(I)=1./(1.-CDF(I))
      RTPK(15)=1000.
      RTPK(14)=500.
      RTPK(13)=200.
      RTPK(12)=100.
      J=1
 301  W=J
      XJ(J)=5.6-W/10.0
      IF(XJ(J)-SK)303,303,302
 302  J=J+1
      GO TO 301
 303  DO 304 I=9,14
      VK =((SK-XJ(J))*(XK(I,J-1)-XK(I,J)))/(XJ(J-1)-XJ(J))+XK(I,J)
      K(I)=EXP(XM+VK*STD)
 304  CONTINUE
      DO 305 I=9,14
 305  Q(I)=K(I)*XMR
      Q10=Q(9)
      Q25=Q(10)
      Q50=Q(11)
      Q100=Q(12)
      Q200=Q(13)
      Q500=Q(14)
C
C     WRITE(6,310)
C310  FORMAT(/' *** LOG PEARSON VARIATE ESTIMATES BY MXM1 METHOD ***'//
C    19X,'CDF',2X,'T(FOR LOS)',2X,'T(FOR PKS)',5X,'VARIATE',//)
C     DO 315 I=1,15
C     WRITE(6,320) CDF(I),RTLF(I),RTPK(I),Q(I)
C315  CONTINUE
C     WRITE(18,330)Q(7),Q(8),Q(9),Q(11),Q(12),Q(13),Q(14)
C330  FORMAT(3X,'MOMIX',1X,7(2X,F7.1))
C320  FORMAT(3F12.3,F12.2)
C     WRITE(6,325)
C325  FORMAT(' NOTE: T=RETURN PERIOD(YRS), LOS=MINIMUM VALUES LIKE LOW F
C    1LOWS, PKS=MAXIMUM VALUES LIKE FLOOD FLOWS')
      RETURN
      END
C
C=======================================================================
C***********************************************************************
C     SUBROUTINE MMDIR1 COMPUTES PARAMETERS BY DIRECT MOM USING THE
C     SCHEME A = F(A)
C***********************************************************************
      SUBROUTINE MMDIR1(A)
      COMMON/LNMMNT/AL1,AL2,AL3
      COMMON/BVAL/B
      COMMON/MMPAR/AA,BETA,C
      REAL L1,L2,L3
      J=1
  20  T1=(ALOG(1.-3.*A)-(3.-2.*B)*ALOG(1.-A))/B
      T2=EXP(T1)
      A1=0.5*(1.0-T2)
      IF(ABS(A1-A).LT.1.0E-6)GO TO 10
      J=J+1
      A=A1
      WRITE(6,*)' A = ',A,' ITERATION NO.  = ',J
      GO TO 20
```

```
  10  AA=A
      BETA=(AL2-2.*AL1)/ALOG((1.-A)*(1.-A)/(1.-2.*A))
      C=AL1+B*ALOG(1.-A)
      WRITE(6,*)' A = ',A,' BETA = ',B,' C = ',C
      RETURN
      END
C
$ENTRY
10 20 30 50 75
//GO.FT09F001 DD DSN=CEAROR.LPT.DATA1,DISP=SHR
//GO.FT10F001 DD DSN=CEAROR.LP3.OUTPUTM1,DISP=SHR
$$
//
```

```
//LGMOM JOB (1304,59634,5,20),'ARORA',MSGCLASS=S,CLASS=A
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=N
//     EXEC WATFIV,REGION.GO=4000K,TIME.GO=99
$JOB           TIME=4500
C
C=================================================================
C                 LOG PEARSON TYPE 3 DISTRIBUTION
C-----------------------------------------------------------------
C     PROGRAM TO COMPUTE PARAMETERS AND QUANTILES BY METHOD OF MOMENT
C     INDIRECT(MMI) APPLIED TO LOG-TRANSFORMED DATA
C=================================================================
C
      COMMON/QUANT/Q10,Q25,Q50,Q100,Q200,Q500
      COMMON/LP3NUM/YR(100100),YR2(100100),YR3(100100)
      COMMON/MMPAR/A,B,C
      COMMON/SIZE/N
      DIMENSION ISSIZE(10)
      DOUBLE PRECISION DSEED
C=================================================================
C       NCSE = NO. OF CASES OF SAMPLE SIZES TO BE ANALYSED
C        M   = NO. OF MONTE-CARLO SAMPLES FOR EACH NCSE
C        N   = SIZE OF EACH SAMPLE (=ISSIZE(.))
C   AP,BP,CP = POPULATION PARAMETERS OF LP3 DISTRIBUTION
C      NRAN  = TOTAL NO. OF MONTE-CARLO NUMBERS GENERATED FOR GIVEN
C              POPULATION PARAMETERS (M*N < NRAN)
C=================================================================
C
      NCSE=5
      READ(5,*)(ISSIZE(I),I=1,NCSE)
      M=1000
C
      NRAN=75000
      DSEED=123457.D0
      AP=0.127683
      BP=10.30311
      CP=-1.407434
      COVP=0.5
      SKP= 3.0
C
      WRITE(6,1)
   1  FORMAT(1H1/'  CASE   2   --   C.V. = 0.5,   SKEW = 3 ')
      WRITE(6,*)
      WRITE(6,*)' POPULATION PARM. : ',AP,BP,CP,'| CV, SKEW :',COVP,SKP
      WRITE(10,*)AP,BP,CP,COVP,SKP,DSEED
C-----------------------------------------------------------------
      CALL LP3GN4(AP,BP,CP,DSEED,NRAN)
C
      DO 10 I=1,NCSE
      N=ISSIZE(I)
      WRITE(6,*)' SAMPLE SIZE = ',N,' NO. OF SAMPLES = ',M
      WRITE(10,*)N,M
C
      DO 20 J=1,M
```

```
      KL=N*(J-1)+1
      KU=N*J
      CALL UBVSK(KL,KU,CPMY,VARL,SKL)
      CALL MMINDR(CPMY,VARL,SKL,A,B,C)
C
      XMR=1.
      STDL=SQRT(VARL)
      IF(ABS(SKL).LE.5.5)GO TO 30
      WRITE(6,*)' LOG SKEW = ',SKL
      GO TO 40
   30 CALL LPQNTL(CPMY,STDL,SKL,XMR)
C
C     CALL SKEWX(A,B,C,SKX)
C     SKXP=SKX/SKP
   40 SKXP=SKL
      WRITE(6,11)J,A,B,C,SKXP,Q10,Q25,Q50,Q100,Q200,Q500
      WRITE(10,11)J,A,B,C,SKXP,Q10,Q25,Q50,Q100,Q200,Q500
   11 FORMAT(1X,I4,F16.7,E15.7,F17.7,F13.7,6F10.5)
   20 CONTINUE
   10 CONTINUE
      STOP
      END
C
C           ********   END OF MAIN SEGMENT   ********
C
C*************************************************************************
C     SUBROUTINE LP3GN4 GENERATES LP3 NUMBERS USING IMSL ROUTINE
C*************************************************************************
      SUBROUTINE LP3GN4(A,B,C,DSEED,NR)
      COMMON/LP3NUM/YR(100100),YR2(100100),YR3(100100)
      DOUBLE PRECISION DSEED
      REAL MR,MR2,MR3
      DIMENSION WK(1)
C
      CALL GGAMR(DSEED,B,NR,WK,YR)
C
      DO 10 I=1,NR
      T=A*YR(I)+C
      YR(I)=T
      YR2(I)=YR(I)*YR(I)
      YR3(I)=YR2(I)*YR(I)
   10 CONTINUE
      RETURN
      END
C*************************************************************************
C    UBVSK: SUB-ROUTINE TO COMPUTE UNBIASED MEAN, VARIANCE, &
C    SKEWNESS COEFFICIENT
C*************************************************************************
      SUBROUTINE UBVSK(KL,KU,VM,VAV,SKV)
      COMMON/LP3NUM/V(100100),V2(100100),V3(100100)
      COMMON/SIZE/N
      FN=FLOAT(N)
      C1=FN/(FN-1.)
      C2=FN**2/(FN-1.)/(FN-2.)
      C2=C2/C1**1.5
      X1=0.
      X2=0.
      X3=0.
      DO 10 I=KL,KU
      X1=X1+V(I)
```

```
      X2=X2+V2(I)
 10   X3=X3+V3(I)
      VM=X1/FN
      VAV=X2/FN-VM**2
      SKV=(X3/FN-3.*VM*VAV-VM**3)/VAV**1.5
      VAV=VAV*C1
      SKV=SKV*C2
C
C     CORR=1.+8.5/FN
C     SKV=SKV*CORR
      RETURN
      END
C*************************************************************************
C     MMINDR : FINDS PARAMETERS BASED ON MEAN,STD. DEV, AND SKEW OF
C              LOG-TRANSFORMED DATA.
C*************************************************************************
      SUBROUTINE MMINDR(CPMY,VARL,SKL,A,B,C)
      B=4./(SKL*SKL)
      A=SQRT(VARL/B)
      IF(SKL.GE.0.0) GO TO 10
      A=-A
  10  C=CPMY-A*B
      RETURN
      END
C
C*************************************************************************
C--LPQNTL - SUB-ROUTINE TO COMPUTE LOG PEARSON QUANTILES. PEARSON
C  FACTORS GIVEN IN WRC BULLETIN #17 (K-TABLES) ARE LINEARLY
C  INTERPOLATED
C*************************************************************************
C
      SUBROUTINE LPQNTL(XM,STD,SK,XMR)
      COMMON/QUANT/Q10,Q25,Q50,Q100,Q200,Q500
      DIMENSION XK(15,111),CDF(15),RTPK(15),RTLF(15),K(15),Q(15),
     1XJ(111),X1(111),X2(111),X3(111),X4(111),X5(111),X6(111),
     1X7(111),X8(111),X9(111),X10(111),X11(111),X12(111),
     1X13(111),X14(111),X15(111)
      REAL K
      DATA CDF/.005,.01,.02,.04,.1,.2,.5,.8,.9,.96,.98,.99,.995,.998,
     1.999/
      DATA X1/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.606,-.625,-.6452,
     3-.6667,-.6896,-.7143,-.7407,-.7691,-.7997,-.8328,-.8686,-.9074,
     4-.9495,-.995,-1.0443,-1.0975,-1.1548,-1.2162,-1.2817,-1.3511,
     5-1.4244,-1.5011,-1.5811,-1.6639,-1.7492,-1.8366,-1.9258,-2.0164,
     6-2.10825,-2.2009,-2.2942,-2.388,-2.4819,-2.5758,
     70.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,
     80.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,25*0./
      DATA X2/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.6061,-.625,-.6451,
     1-0.6666,-0.6896,-.7145,-0.7405,-0.7688,-0.7992,-0.832,-0.8672,
     1-0.9052,-0.9461,-0.99,-1.037,-1.0871,-1.1404,-1.1968,-1.2561,
     2-1.3182,-1.3827,-1.4494,-1.5181,-1.5884,-1.66,-1.7327,-1.8062,
     3-1.8803,-1.9547,-2.0293,-2.1039,-2.1784,-2.2526,-2.3264,55*0./
      DATA X3/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.6061,-.625,-.6451,
     1-0.6665,-0.6894,-.7138,-0.7399,-0.7678,-0.79765,-0.8296,-0.8637,
```

```
1-0.9001,-0.9388,-0.9798,-1.0231,-1.0686,-1.1163,-1.1658,-1.2172,
1-1.27,-1.3241,-1.3793,-1.4353,-1.4919,-1.5489,-1.606,-1.6633,
2-1.7203,-1.7772,-1.8336,-1.8896,-1.945,-1.9997,-2.0538,55*0./
 DATA X5/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
1-.4255,-.4348,-.44443,-.45452,-.4651,-.4761,-.4877,-.4999,
2-.5126,-.526,-.5401,-.5548,-.57035,-.5867,-.6038,-.62175,-.6406,
1-0.6602,-0.6808,-.7021,-0.7242,-0.7471,-0.7706,-0.7947,
1-0.8193,-0.8442,-0.8694,-0.8946,-0.9199,-0.945,-0.9698,-0.9942,
2-1.0181,-1.0414,-1.0641,-1.0861,-1.1073,-1.1276,-1.1471,-1.1657,
3-1.1835,-1.2003,-1.2162,-1.2311,-1.2452,-1.2582,-1.2704,-1.2816,
455*0./
 DATA X6/-.3636,-.3704,-.37734,-.38458,-.39211,-.39993,-.40806,
1-.4165,-.4253,-.4345,-.444,-.454,-.4643,-.475,-.4862,-.4978,
2-.5099,-.5224,-.5353,-.5487,-.5624,-.5765,-.591,-.6057,-.6206,
1-0.6357,-0.6509,-.666,-0.6811,-0.696,-0.7107,-0.725,-0.7388,
1-0.7521,-0.7648,-0.7769,-0.7882,-0.7987,-0.8084,-0.8172,-0.8252,
2-.8322,-.8384,-.8437,-.8481,-.8516,-.8543,-.8561,-.857,-.8572,
3-.8565,-.8551,-.8529,-.8499,-.8461,-.8416,55*0./
 DATA X7/-.3546,-.3596,-.3645,-.3695,-.3743,-.379,-.3836,-.388,
1-.3922,-.3962,-.3999,-.4032,-.4062,-.4088,-.411,-.4127,-.4138,
2-.4144,-.4144,-.4138,-.4125,-.4106,-.4079,-.4045,-.4004,
1-.3955,-.3899,-.3835,-.3764,-.3685,-.3599,-.3506,-.3406,
1-.33,-.3187,-.3069,-.2944,-.2815,-.2681,-.2542,-.24,-.2254,
2-.2104,-.1952,-.1797,-.164,-.1481,-.132,-.1158,-.0995,-.083,
3-.0665,-.0499,-.0333,-.0166,0.,55*0./
 DATA X8/-.0103,.00243,.0156,.0293,.0434,.058,.073,.0885,.1044,
1.1207,.1374,.1545,.1719,.1897,.2078,.2262,.2448,.2638,.2829,.3022,
2.3217,.3413,.361,.3808,.4006,
1.4204,.4402,.4598,.4793,.4987,.5179,.5368,.5555,.5738,
1.5918,.6094,.6266,.6434,.6596,.6753,.6905,.7051,.7192,.7326,
2.7454,.7575,.769,.7799,.79,.7995,.8083,.8164,.8238,.8304,.8364,
3.8416,55*0./
 DATA X9/.6912,.712,.7328,.7536,.7746,.7955,.8164,.8373,.8582,
1.879,.8996,.9202,.9406,.9609,.981,1.0008,1.0204,1.0397,1.0586,
21.0773,1.0955,1.1134,1.1308,1.1477,1.1642,1.1801,1.1954,
11.2101,1.2242,1.2377,1.2504,1.2624,1.2737,1.2841,1.2938,
11.3026,1.3105,1.3176,1.3238,1.329,1.3333,1.3367,1.339,1.3405,
21.3409,1.3404,1.3389,1.3364,1.3329,1.3285,1.3231,1.3167,1.3094,
31.3011,1.2918,1.2816,55*0./
 DATA X10/2.0474,2.0637,2.0795,2.0949,2.1099,2.1243,2.1383,2.1517,
12.1647,2.177,2.1887,2.1999,2.2104,2.2202,2.2294,2.2379,2.2456,
22.2525,2.2587,2.2641,2.2686,2.2723,2.2751,2.2769,2.2779,2.2778,
12.27676,2.2747,2.2716,2.2674,2.2622,2.2558,2.2483,2.2397,
12.2299,2.2189,2.2067,2.1933,2.1787,2.1629,2.1459,2.1277,2.1082,
22.0876,2.0657,2.0427,2.0185,1.9931,1.9666,1.939,1.9102,1.8804,
31.8495,1.8176,1.7846,1.7507,1.7158,1.68,1.6433,1.6057,1.5674,
41.5283,1.4885,1.4481,1.4072,1.3658,1.3241,1.2823,1.2403,1.1984,
51.1568,1.1157,1.0751,1.0354,.9967,.9592,.923,.8881,.8549,.8232,
6.7931,.7646,.7377,.7123,.6884,.6659,
6.6447,.6247,.6059,.5881,.5714,.5555,.5405,.5263,.5128,.5,.4878,
7.4762,.4651,.4546,.4444,.4348,.4255,.4167,.4082,.4,.3922,.3846,
8.3774,.3704,.3636/
 DATA X11/3.2838,3.2884,3.2924,3.2957,3.2982,3.30007,3.3012,
13.3015,3.301,3.2998,3.2977,3.2947,3.2909,3.2862,3.2806,3.274,
23.2665,3.258,3.2485,3.238,3.2264,3.2138,3.2,3.1851,3.1691,
33.1519,3.1336,3.114,3.0932,3.0712,3.0479,3.0233,2.9974,2.9703,
42.9418,2.912,2.8809,2.8485,2.8147,2.7796,2.7433,2.7056,2.6666,
52.6263,2.5848,2.5421,2.4981,2.453,2.4067,2.3593,2.3108,2.2613,
62.2108,2.1594,2.107,2.0538,55*0./
 DATA X12/4.6402,4.6285,4.6159,4.6025,4.5882,4.573,4.5569,4.5399,
```

```
     14.5219,4.503,4.483,4.4621,4.4401,4.4171,4.393,4.3678,4.3415,4.314,
     24.2855,4.2557,4.2247,4.1926,4.1592,4.1245,4.0886,
     34.0514,4.0129,3.973,3.9318,3.8893,3.8454,3.8001,3.7535,3.7054,
     43.656,3.6052,3.553,3.4994,3.4444,3.388,3.3304,3.2713,3.211,
     53.1494,3.0866,3.0226,2.9574,2.891,2.8236,2.7551,2.6857,2.6154,
     62.5442,2.4723,2.3996,2.3264,55*0./
      DATA X13/6.08307,6.0517,6.0193,5.986,5.9517,5.9164,5.88,5.8427,
     15.8042,5.7646,5.724,5.6822,5.6393,5.5953,5.5501,5.5036,5.456,
     25.4071,5.357,5.3056,5.2529,5.1989,5.1436,5.087,5.029,
     34.9696,4.9088,4.8467,4.7831,4.7182,4.6518,4.5839,4.5147,4.444,
     44.3719,4.2983,4.2234,4.147,4.0693,3.9902,3.9097,3.828,3.745,
     53.6607,3.5753,3.4887,3.4011,3.3124,3.2228,3.1323,3.041,2.949,
     62.8564,2.7632,2.6697,2.5758,55*0./
      DATA X14/8.0869,8.0259,7.9639,7.9008,7.8366,7.7712,7.7048,
     17.6372,7.5684,7.4985,7.4273,7.355,7.2814,7.2065,7.1304,7.053,
     26.9744,6.8944,6.813,6.7303,6.6463,6.5608,6.474,6.3858,6.2961,
     16.20506,6.1125,6.0186,5.9232,5.8263,5.728,5.6282,5.5269,
     15.4243,5.3201,5.2146,5.1077,4.9994,4.8897,4.7788,4.6665,4.553,
     24.4384,4.3226,4.2058,4.088,3.9693,3.8498,3.7296,3.6087,3.4874,
     33.3657,3.2437,3.1217,2.9998,2.8782,2.7571,2.6367,2.5174,
     42.3994,2.2831,2.1688,2.057,1.9481,1.8424,1.7406,1.6431,1.5502,
     51.4623,1.3798,1.3028,1.2313,1.1653,1.1047,1.049,.998,.9513,.9085,
     6.8693,.8332,.7999,.7692,.7407,.7143,.6896,.6667,25*0./
      DATA X15/9.6577,9.5723,9.4859,9.3983,9.3095,9.2196,9.1285,9.0362,
     18.9427,8.848,8.752,8.6548,8.5563,8.4565,8.3553,8.2529,8.1491,
     28.044,7.9374,7.8295,7.7202,7.6095,7.4974,7.3838,7.2688,
     17.1524,7.0344,6.9151,6.7942,6.6719,6.5481,6.4229,6.2963,
     26.1682,6.0387,5.9078,5.7755,5.6419,5.507,5.3709,5.2335,5.0951,
     34.9555,4.8149,4.6734,4.5311,4.3881,4.2444,4.1002,3.9557,3.8109,
     43.6661,3.5214,3.377,3.2332,3.0902,2.9483,2.8079,2.6692,2.5326,
     52.3987,2.2678,2.1405,2.0174,1.8989,1.7857,1.6783,1.577,1.4822,
     61.3941,1.3128,1.2381,1.1697,1.1074,1.0507,.999,.9519,.9089,.8695,
     7.8333,.8,.7692,.7407,.7143,.6897,.6667,25*0./
      J=111
      DO 61 I=1,25
      X14(J)=-X1(I)
      X15(J)=-X1(I)
61    J=J-1
      J=111
      DO 62 I=1,111
      X4(J)=-X10(I)
62    J=J-1
      J=111
      DO 63 I=1,55
      X1(J)=-X13(I)
      X2(J)=-X12(I)
      X3(J)=-X11(I)
      X5(J)=-X9(I)
      X6(J)=-X8(I)
      X7(J)=-X7(I)
      X8(J)=-X6(I)
      X9(J)=-X5(I)
      X11(J)=-X3(I)
      X12(J)=-X2(I)
      X13(J)=-X1(I)
63    J=J-1
      DO 1 J =1,111
      XK(1,J)=X1(J)
      XK(2,J)=X2(J)
      XK(3,J)=X3(J)
```

```
      XK(4,J)=X4(J)
      XK(5,J)=X5(J)
      XK(6,J)=X6(J)
      XK(7,J)=X7(J)
      XK(8,J)=X8(J)
      XK(9,J)=X9(J)
      XK(10,J)=X10(J)
      XK(11,J)=X11(J)
      XK(12,J)=X12(J)
      XK(13,J)=X13(J)
      XK(14,J)=X14(J)
      XK(15,J)=X15(J)
    1 CONTINUE
      DO 65 I=1,15
      RTLF(I)=1./CDF(I)
   65 RTPK(I)=1./(1.-CDF(I))
      RTPK(15)=1000.
      RTPK(14)=500.
      RTPK(13)=200.
      RTPK(12)=100.
      J=1
  301 W=J
      XJ(J)=5.6-W/10.0
      IF(XJ(J)-SK)303,303,302
  302 J=J+1
      GO TO 301
  303 DO 304 I=9,14
      VK =((SK-XJ(J))*(XK(I,J-1)-XK(I,J)))/(XJ(J-1)-XJ(J))+XK(I,J)
      K(I)=EXP(XM+VK*STD)
  304 CONTINUE
      DO 305 I=9,14
  305 Q(I)=K(I)*XMR
      Q10=Q(9)
      Q25=Q(10)
      Q50=Q(11)
      Q100=Q(12)
      Q200=Q(13)
      Q500=Q(14)
C
C     WRITE(6,310)
C310  FORMAT(/' *** LOG PEARSON VARIATE ESTIMATES BY MXM1 METHOD ***'//
C    19X,'CDF',2X,'T(FOR LOS)',2X,'T(FOR PKS)',5X,'VARIATE',//)
C     DO 315 I=1,15
C     WRITE(6,320) CDF(I),RTLF(I),RTPK(I),Q(I)
C315  CONTINUE
C     WRITE(18,330)Q(7),Q(8),Q(9),Q(11),Q(12),Q(13),Q(14)
C330  FORMAT(3X,'MOMIX',1X,7(2X,F7.1))
C320  FORMAT(3F12.3,F12.2)
C     WRITE(6,325)
C325  FORMAT(' NOTE: T=RETURN PERIOD(YRS), LOS=MINIMUM VALUES LIKE LOW F
C    1LOWS, PKS=MAXIMUM VALUES LIKE FLOOD FLOWS')
      RETURN
      END
$ENTRY
10 20 30 50 75
//GO.FT10F001 DD DSN=CEAROR.LP3.OUTPUTM3,DISP=SHR
$$
//
```

```
//MIXCASE2 JOB (1304,59634,6,20),'ARORA',MSGCLASS=S,CLASS=A
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=D
//     EXEC WATFIV,REGION.GO=4000K,TIME.GO=99
$JOB            TIME=4500
C
C=================================================================
C                  LOG PEARSON TYPE 3 DISTRIBUTION
C-----------------------------------------------------------------
C      PROGRAM TO COMPUTE PARAMETERS AND QUANTILES BY METHOD OF MIXED
C      MOMENTS (MIX)
C=================================================================
C
       COMMON/QUANT/Q10,Q25,Q50,Q100,Q200,Q500
       COMMON/LP3NUM/R(100100),R2(100100)
       COMMON/ALL/PTAB(345),ATAB(345)
       COMMON/MMPAR/A,B,C
       COMMON/SIZE/N
       DIMENSION ISSIZE(10)
       DOUBLE PRECISION DSEED
C=================================================================
C        NCSE = NO. OF CASES OF SAMPLE SIZES TO BE ANALYSED
C         M   = NO. OF MONTE-CARLO SAMPLES FOR EACH NCSE
C         N   = SIZE OF EACH SAMPLE (=ISSIZE(.))
C    AP,BP,CP = POPULATION PARAMETERS OF LP3 DISTRIBUTION
C       NRAN  = TOTAL NO. OF MONTE-CARLO NUMBERS GENERATED FOR GIVEN
C               POPULATION PARAMETERS (M*N < NRAN)
C=================================================================
C
       NCSE=5
       READ(5,*)(ISSIZE(I),I=1,NCSE)
       READ(9,*)(PTAB(I),ATAB(I),I=1,345)
       DO 100 I=1,345
  100  PTAB(I)=-PTAB(I)
C
       M=1000
C
       NRAN=75000
       DSEED=123457.D0
       AP=0.059798
       BP=98.38009
       CP=-6.066213
       COVP=0.7
       SKP= 3.0
C
       WRITE(6,1)
    1  FORMAT(1H1/'  CASE   5   --   C.V. = 0.7,   SKEW = 3 ')
       WRITE(6,*)
       WRITE(6,*)' POPULATION PARM. : ',AP,BP,CP,'| CV, SKEW :',COVP,SKP
       WRITE(10,*)AP,BP,CP,COVP,SKP,DSEED
C-----------------------------------------------------------------
       CALL LP3GN4(AP,BP,CP,DSEED,NRAN)
C
       DO 10 I=1,NCSE
```

```
      N=ISSIZE(I)
      WRITE(6,*)' SAMPLE SIZE = ',N,' NO. OF SAMPLES = ',M
      WRITE(10,*)N,M
C
      DO 20 J=1,M
      KL=N*(J-1)+1
      KU=N*J
      CALL STATS(KL,KU)
      CALL POLATE
      CALL MIX
C
      XMR=1.
      CPMY=C+A*B
      VARL=B*A*A
      STDL=SQRT(VARL)
      SKL=2.*(ABS(A)/A)/SQRT(B)
C
      IF(ABS(SKL).LE.5.5)GO TO 30
      WRITE(6,*)' LOG SKEW = ',SKL
      GO TO 40
C
   30 CALL LPQNTL(CPMY,STDL,SKL,XMR)
C
C     CALL SKEWX(A,B,C,SKX)
C     SKXP=SKX/SKP
   40 SKXP=SKL
      WRITE(6,11)J,A,B,C,SKXP,Q10,Q25,Q50,Q100,Q200,Q500
      WRITE(10,11)J,A,B,C,SKXP,Q10,Q25,Q50,Q100,Q200,Q500
   11 FORMAT(1X,I4,F16.7,E15.7,F17.7,F13.7,6F10.5)
   20 CONTINUE
   10 CONTINUE
      STOP
      END
C
C           ********   END OF MAIN SEGMENT   ********
C
C*************************************************************************
C     SUBROUTINE LP3GN4 GENERATES LP3 NUMBERS USING IMSL ROUTINE
C*************************************************************************
      SUBROUTINE LP3GN4(A,B,C,DSEED,NR)
      COMMON/LP3NUM/R(100100),R2(100100)
      COMMON/YLP3/YR(100100)
      DOUBLE PRECISION DSEED
      DOUBLE PRECISION DT,DR,DDR
      DIMENSION WK(1)
C
      CALL GGAMR(DSEED,B,NR,WK,R)
C
      DO 10 I=1,NR
      T=A*R(I)+C
      DT=T
      DR=DEXP(DT)
      R(I)=DR
      DDR=DLOG(DR)
      YR(I)=DDR
      R2(I)=R(I)*R(I)
   10 CONTINUE
      RETURN
      END
C*************************************************************************
```

```
C     SUBROUTINE STATS COMPUTES THE SAMPLE DEPENDENT VALUE OF P
C     AS A FUNCTION OF MEAN, VARIANCE OF REAL DATA, AND MEAN OF
C     LOG-TRANSFORMED DATA.
C*************************************************************************
      SUBROUTINE STATS(KL,KU)
      COMMON/YLP3/YR(100100)
      COMMON/PVAL/P
      COMMON/SIZE/N
      COMMON/NUM/DYM,TM
      DOUBLE PRECISION DXM,DVAX,DYM,TN,TM,DP
C
      CALL UBVSK(KL,KU,XM,VAX)
      DXM=XM
      DVAX=VAX
C
      FN=FLOAT(N)
      SUM=0.0
      DO 10 I=KL,KU
 10   SUM=SUM+YR(I)
      YM=SUM/FN
      DYM=YM
C
      TN=DVAX/(DXM*DXM)+1.D0
      TN=DLOG(TN)
      TM=DYM-DLOG(DXM)
      DP=TN/TM
      P=DP
      RETURN
      END
C*************************************************************************
C     UBVSK: SUB-ROUTINE TO COMPUTE UNBIASED MEAN, VARIANCE, &
C     SKEWNESS COEFFICIENT
C*************************************************************************
      SUBROUTINE UBVSK(KL,KU,VM,VAV)
      COMMON/LP3NUM/V(100100),V2(100100)
      COMMON/SIZE/N
      FN=FLOAT(N)
      C1=FN/(FN-1.)
      X1=0.
      X2=0.
      DO 10 I=KL,KU
      X1=X1+V(I)
      X2=X2+V2(I)
 10   CONTINUE
      VM=X1/FN
      VAV=X2/FN-VM**2
      VAV=VAV*C1
      RETURN
      END
C*************************************************************************
C     SUBROUTINE POLATE INTERPOLATES FOR METHOD OF MIXED MOMENTS
C     USING PTAB(.),ATAB(.)
C*************************************************************************
      SUBROUTINE POLATE
      COMMON/PVAL/P
      COMMON/EST/AEST
      COMMON/ALL/PTAB(345),ATAB(345)
      DO 10 I= 1,345
      IF((P.GT.-0.0016244).OR. (P .LT.-28.669230))GO TO 12
      IF((P.LE.PTAB(I)).AND. (P .GE. PTAB(I+1)))GO TO 50
```

```
 10    CONTINUE
 50    DELALP= (ATAB(I)-ATAB(I+1))/(PTAB(I)-PTAB(I+1))
      1*(P-PTAB(I+1))
       AEST= DELALP+ATAB(I+1)
       RETURN
 12    WRITE(6,3)
  3    FORMAT(1X,'NO DIRECT MOMENT SOLUTION POSSIBLE')
       RETURN
       END
C*************************************************************************
C      MIX : COMPUTES THE PARAMETERS BY MIXED MOMENT METHOD
C*************************************************************************
       SUBROUTINE MIX
       COMMON/MMPAR/AS,BS,CS
       COMMON/PVAL/P
       COMMON/EST/AEST
       COMMON/NUM/DYM,TM
       DOUBLE PRECISION DYM,TM,A,B,C,A1,AL1
C
       A=AEST
       CALL ROOT(A)
C
       A1=1.D0-A
       AL1=DLOG(A1)
C
       B=TM/(AL1+A)
       C=DYM-A*B
C
       AS=A
       BS=B
       CS=C
       RETURN
       END
C*************************************************************************
C ROOT : REFINES THE INTERPOLATED VALUE OF PARAMETER A BY NEWTON RAPHSON
C*************************************************************************
       SUBROUTINE ROOT(A)
       COMMON/PVAL/PSAMPL
       DOUBLE PRECISION A,A1,A2,AL1,AL2,F,DF,P,PP,DA,ANEW
       P=PSAMPL
       PP=2.D0-P
       ITR=1
 20    A1=1.D0-A
       A2=1.D0-2.D0*A
       AL1=DLOG(A1)
       AL2=DLOG(A2)
       F=-AL2+PP*AL1-A*P
       DF=2.D0/A2-PP/A1-P
       DA=F/DF
       ANEW=A-DA
C      WRITE(6,*)' ROOTS OF A = ',ITR,A,ANEW
       IF(DABS(ANEW-A).LT.1.0D-08)GO TO 30
       IF(ITR.GT.100)GO TO 30
       ITR=ITR+1
       A=ANEW
       GO TO 20
 30    A=ANEW
       RETURN
       END
C*************************************************************************
```

```
C--LPQNTL - SUB-ROUTINE TO COMPUTE LOG PEARSON QUANTILES. PEARSON
C  FACTORS GIVEN IN WRC BULLETIN #17 (K-TABLES) ARE LINEARLY
C  INTERPOLATED
C**********************************************************************
C
      SUBROUTINE LPQNTL(XM,STD,SK,XMR)
      COMMON/QUANT/Q10,Q25,Q50,Q100,Q200,Q500
      DIMENSION XK(15,111),CDF(15),RTPK(15),RTLF(15),K(15),Q(15),
     1XJ(111),X1(111),X2(111),X3(111),X4(111),X5(111),X6(111),
     1X7(111),X8(111),X9(111),X10(111),X11(111),X12(111),
     1X13(111),X14(111),X15(111)
      REAL K
      DATA CDF/.005,.01,.02,.04,.1,.2,.5,.8,.9,.96,.98,.99,.995,.998,
     1.999/
      DATA X1/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.606,-.625,-.6452,
     3-.6667,-.6896,-.7143,-.7407,-.7691,-.7997,-.8328,-.8686,-.9074,
     4-.9495,-.995,-1.0443,-1.0975,-1.1548,-1.2162,-1.2817,-1.3511,
     5-1.4244,-1.5011,-1.5811,-1.6639,-1.7492,-1.8366,-1.9258,-2.0164,
     6-2.10825,-2.2009,-2.2942,-2.388,-2.4819,-2.5758,
     70.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,
     80.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,25*0./
      DATA X2/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.6061,-.625,-.6451,
     1-0.6666,-0.6896,-.7145,-0.7405,-0.7688,-0.7992,-0.832,-0.8672,
     1-0.9052,-0.9461,-0.99,-1.037,-1.0871,-1.1404,-1.1968,-1.2561,
     2-1.3182,-1.3827,-1.4494,-1.5181,-1.5884,-1.66,-1.7327,-1.8062,
     3-1.8803,-1.9547,-2.0293,-2.1039,-2.1784,-2.2526,-2.3264,55*0./
      DATA X3/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.6061,-.625,-.6451,
     1-0.6665,-0.6894,-.7138,-0.7399,-0.7678,-0.79765,-0.8296,-0.8637,
     1-0.9001,-0.9388,-0.9798,-1.0231,-1.0686,-1.1163,-1.1658,-1.2172,
     1-1.27,-1.3241,-1.3793,-1.4353,-1.4919,-1.5489,-1.606,-1.6633,
     2-1.7203,-1.7772,-1.8336,-1.8896,-1.945,-1.9997,-2.0538,55*0./
      DATA X5/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44443,-.45452,-.4651,-.4761,-.4877,-.4999,
     2-.5126,-.526,-.5401,-.5548,-.57035,-.5867,-.6038,-.62175,-.6406,
     1-0.6602,-0.6808,-.7021,-0.7242,-0.7471,-0.7706,-0.7947,
     1-0.8193,-0.8442,-0.8694,-0.8946,-0.9199,-0.945,-0.9698,-0.9942,
     2-1.0181,-1.0414,-1.0641,-1.0861,-1.1073,-1.1276,-1.1471,-1.1657,
     3-1.1835,-1.2003,-1.2162,-1.2311,-1.2452,-1.2582,-1.2704,-1.2816,
     455*0./
      DATA X6/-.3636,-.3704,-.37734,-.38458,-.39211,-.39993,-.40806,
     1-.4165,-.4253,-.4345,-.444,-.454,-.4643,-.475,-.4862,-.4978,
     2-.5099,-.5224,-.5353,-.5487,-.5624,-.5765,-.591,-.6057,-.6206,
     1-0.6357,-0.6509,-.666,-0.6811,-0.696,-0.7107,-0.725,-0.7388,
     1-0.7521,-0.7648,-0.7769,-0.7882,-0.7987,-0.8084,-0.8172,-0.8252,
     2-.8322,-.8384,-.8437,-.8481,-.8516,-.8543,-.8561,-.857,-.8572,
     3-.8565,-.8551,-.8529,-.8499,-.8461,-.8416,55*0./
      DATA X7/-.3546,-.3596,-.3645,-.3695,-.3743,-.379,-.3836,-.388,
     1-.3922,-.3962,-.3999,-.4032,-.4062,-.4088,-.411,-.4127,-.4138,
     2-.4144,-.4144,-.4138,-.4125,-.4106,-.4079,-.4045,-.4004,
     1-.3955,-.3899,-.3835,-.3764,-.3685,-.3599,-.3506,-.3406,
     1-.33,-.3187,-.3069,-.2944,-.2815,-.2681,-.2542,-.24,-.2254,
     2-.2104,-.1952,-.1797,-.164,-.1481,-.132,-.1158,-.0995,-.083,
     3-.0665,-.0499,-.0333,-.0166,0.,55*0./
      DATA X8/-.0103,.00243,.0156,.0293,.0434,.058,.073,.0885,.1044,
```

```
1.1207,.1374,.1545,.1719,.1897,.2078,.2262,.2448,.2638,.2829,.3022,
2.3217,.3413,.361,.3808,.4006,
1.4204,.4402,.4598,.4793,.4987,.5179,.5368,.5555,.5738,
1.5918,.6094,.6266,.6434,.6596,.6753,.6905,.7051,.7192,.7326,
2.7454,.7575,.769,.7799,.79,.7995,.8083,.8164,.8238,.8304,.8364,
3.8416,55*0./
 DATA X9/.6912,.712,.7328,.7536,.7746,.7955,.8164,.8373,.8582,
1.879,.8996,.9202,.9406,.9609,.981,1.0008,1.0204,1.0397,1.0586,
21.0773,1.0955,1.1134,1.1308,1.1477,1.1642,1.1801,1.1954,
11.2101,1.2242,1.2377,1.2504,1.2624,1.2737,1.2841,1.2938,
11.3026,1.3105,1.3176,1.3238,1.329,1.3333,1.3367,1.339,1.3405,
21.3409,1.3404,1.3389,1.3364,1.3329,1.3285,1.3231,1.3167,1.3094,
31.3011,1.2918,1.2816,55*0./
 DATA X10/2.0474,2.0637,2.0795,2.0949,2.1099,2.1243,2.1383,2.1517,
12.1647,2.177,2.1887,2.1999,2.2104,2.2202,2.2294,2.2379,2.2456,
22.2525,2.2587,2.2641,2.2686,2.2723,2.2751,2.2769,2.2779,2.2778,
12.27676,2.2747,2.2716,2.2674,2.2622,2.2558,2.2483,2.2397,
12.2299,2.2189,2.2067,2.1933,2.1787,2.1629,2.1459,2.1277,2.1082,
22.0876,2.0657,2.0427,2.0185,1.9931,1.9666,1.939,1.9102,1.8804,
31.8495,1.8176,1.7846,1.7507,1.7158,1.68,1.6433,1.6057,1.5674,
41.5283,1.4885,1.4481,1.4072,1.3658,1.3241,1.2823,1.2403,1.1984,
51.1568,1.1157,1.0751,1.0354,.9967,.9592,.923,.8881,.8549,.8232,
6.7931,.7646,.7377,.7123,.6884,.6659,
6.6447,.6247,.6059,.5881,.5714,.5555,.5405,.5263,.5128,.5,.4878,
7.4762,.4651,.4546,.4444,.4348,.4255,.4167,.4082,.4,.3922,.3846,
8.3774,.3704,.3636/
 DATA X11/3.2838,3.2884,3.2924,3.2957,3.2982,3.30007,3.3012,
13.3015,3.301,3.2998,3.2977,3.2947,3.2909,3.2862,3.2806,3.274,
23.2665,3.258,3.2485,3.238,3.2264,3.2138,3.2,3.1851,3.1691,
33.1519,3.1336,3.114,3.0932,3.0712,3.0479,3.0233,2.9974,2.9703,
42.9418,2.912,2.8809,2.8485,2.8147,2.7796,2.7433,2.7056,2.6666,
52.6263,2.5848,2.5421,2.4981,2.453,2.4067,2.3593,2.3108,2.2613,
62.2108,2.1594,2.107,2.0538,55*0./
 DATA X12/4.6402,4.6285,4.6159,4.6025,4.5882,4.573,4.5569,4.5399,
14.5219,4.503,4.483,4.4621,4.4401,4.4171,4.393,4.3678,4.3415,4.314,
24.2855,4.2557,4.2247,4.1926,4.1592,4.1245,4.0886,
34.0514,4.0129,3.973,3.9318,3.8893,3.8454,3.8001,3.7535,3.7054,
43.656,3.6052,3.553,3.4994,3.4444,3.388,3.3304,3.2713,3.211,
53.1494,3.0866,3.0226,2.9574,2.891,2.8236,2.7551,2.6857,2.6154,
62.5442,2.4723,2.3996,2.3264,55*0./
 DATA X13/6.08307,6.0517,6.0193,5.986,5.9517,5.9164,5.88,5.8427,
15.8042,5.7646,5.724,5.6822,5.6393,5.5953,5.5501,5.5036,5.456,
25.4071,5.357,5.3056,5.2529,5.1989,5.1436,5.087,5.029,
34.9696,4.9088,4.8467,4.7831,4.7182,4.6518,4.5839,4.5147,4.444,
44.3719,4.2983,4.2234,4.147,4.0693,3.9902,3.9097,3.828,3.745,
53.6607,3.5753,3.4887,3.4011,3.3124,3.2228,3.1323,3.041,2.949,
62.8564,2.7632,2.6697,2.5758,55*0./
 DATA X14/8.0869,8.0259,7.9639,7.9008,7.8366,7.7712,7.7048,
17.6372,7.5684,7.4985,7.4273,7.355,7.2814,7.2065,7.1304,7.053,
26.9744,6.8944,6.813,6.7303,6.6463,6.5608,6.474,6.3858,6.2961,
16.20506,6.1125,6.0186,5.9232,5.8263,5.728,5.6282,5.5269,
15.4243,5.3201,5.2146,5.1077,4.9994,4.8897,4.7788,4.6665,4.553,
24.4384,4.3226,4.2058,4.088,3.9693,3.8498,3.7296,3.6087,3.4874,
33.3657,3.2437,3.1217,2.9998,2.8782,2.7571,2.6367,2.5174,
42.3994,2.2831,2.1688,2.057,1.9481,1.8424,1.7406,1.6431,1.5502,
51.4623,1.3798,1.3028,1.2313,1.1653,1.1047,1.049,.998,.9513,.9085,
6.8693,.8332,.7999,.7692,.7407,.7143,.6896,.6667,25*0./
 DATA X15/9.6577,9.5723,9.4859,9.3983,9.3095,9.2196,9.1285,9.0362,
18.9427,8.848,8.752,8.6548,8.5563,8.4565,8.3553,8.2529,8.1491,
28.044,7.9374,7.8295,7.7202,7.6095,7.4974,7.3838,7.2688,
```

```
     17.1524,7.0344,6.9151,6.7942,6.6719,6.5481,6.4229,6.2963,
     26.1682,6.0387,5.9078,5.7755,5.6419,5.507,5.3709,5.2335,5.0951,
     34.9555,4.8149,4.6734,4.5311,4.3881,4.2444,4.1002,3.9557,3.8109,
     43.6661,3.5214,3.377,3.2332,3.0902,2.9483,2.8079,2.6692,2.5326,
     52.3987,2.2678,2.1405,2.0174,1.8989,1.7857,1.6783,1.577,1.4822,
     61.3941,1.3128,1.2381,1.1697,1.1074,1.0507,.999,.9519,.9089,.8695,
     7.8333,.8,.7692,.7407,.7143,.6897,.6667,25*0./
      J=111
      DO 61 I=1,25
      X14(J)=-X1(I)
      X15(J)=-X1(I)
61    J=J-1
      J=111
      DO 62 I=1,111
      X4(J)=-X10(I)
62    J=J-1
      J=111
      DO 63 I=1,55
      X1(J)=-X13(I)
      X2(J)=-X12(I)
      X3(J)=-X11(I)
      X5(J)=-X9(I)
      X6(J)=-X8(I)
      X7(J)=-X7(I)
      X8(J)=-X6(I)
      X9(J)=-X5(I)
      X11(J)=-X3(I)
      X12(J)=-X2(I)
      X13(J)=-X1(I)
63    J=J-1
      DO 1 J =1,111
      XK(1,J)=X1(J)
      XK(2,J)=X2(J)
      XK(3,J)=X3(J)
      XK(4,J)=X4(J)
      XK(5,J)=X5(J)
      XK(6,J)=X6(J)
      XK(7,J)=X7(J)
      XK(8,J)=X8(J)
      XK(9,J)=X9(J)
      XK(10,J)=X10(J)
      XK(11,J)=X11(J)
      XK(12,J)=X12(J)
      XK(13,J)=X13(J)
      XK(14,J)=X14(J)
      XK(15,J)=X15(J)
 1    CONTINUE
      DO 65 I=1,15
      RTLF(I)=1./CDF(I)
65    RTPK(I)=1./(1.-CDF(I))
      RTPK(15)=1000.
      RTPK(14)=500.
      RTPK(13)=200.
      RTPK(12)=100.
      J=1
301   W=J
      XJ(J)=5.6-W/10.0
      IF(XJ(J)-SK)303,303,302
302   J=J+1
      GO TO 301
```

```
 303  DO 304 I=9,14
      VK =((SK-XJ(J))*(XK(I,J-1)-XK(I,J)))/(XJ(J-1)-XJ(J))+XK(I,J)
      K(I)=EXP(XM+VK*STD)
 304  CONTINUE
      DO 305 I=9,14
 305  Q(I)=K(I)*XMR
      Q10=Q(9)
      Q25=Q(10)
      Q50=Q(11)
      Q100=Q(12)
      Q200=Q(13)
      Q500=Q(14)
C
C     WRITE(6,310)
C310  FORMAT(/' *** LOG PEARSON VARIATE ESTIMATES BY MXM1 METHOD ***'//
C    19X,'CDF',2X,'T(FOR LOS)',2X,'T(FOR PKS)',5X,'VARIATE',//)
C     DO 315 I=1,15
C     WRITE(6,320) CDF(I),RTLF(I),RTPK(I),Q(I)
C315  CONTINUE
C     WRITE(18,330)Q(7),Q(8),Q(9),Q(11),Q(12),Q(13),Q(14)
C330  FORMAT(3X,'MOMIX',1X,7(2X,F7.1))
C320  FORMAT(3F12.3,F12.2)
C     WRITE(6,325)
C325  FORMAT(' NOTE: T=RETURN PERIOD(YRS), LOS=MINIMUM VALUES LIKE LOW F
C    1LOWS, PKS=MAXIMUM VALUES LIKE FLOOD FLOWS')
      RETURN
      END
$ENTRY
10 20 30 50 75
//GO.FT09F001 DD DSN=CEAROR.LPT.DATAMIX,DISP=SHR
//GO.FT10F001 DD DSN=CEAROR.LP3.OUTPUTM5,DISP=SHR
$$
//
```

```
//MLEC575 JOB (1304,59634,50,20),'ARORA',MSGCLASS=S,CLASS=H
/*ROUTE PRINT CEBA
//     EXEC WATFIV,LIB='CEAROR.SPEC.LIB',REGION.GO=4000K,TIME.GO=99
$JOB            TIME=4500,NOEXT
C
C=================================================================
C                  LOG PEARSON TYPE 3 DISTRIBUTION
C-----------------------------------------------------------------
C     PROGRAM TO COMPUTE PARAMETERS AND QUANTILES BY THE METHOD OF
C     MAXIMUM LIKELIHOOD ESTIMATION (MLE).
C=================================================================
C
C/*JOBPARM SHIFT=N
      DOUBLE PRECISION DSEED,DFN,YK
      DOUBLE PRECISION FLLD1,FLLD2,FLLD2N
      COMMON/P3YK/YK(100)
      COMMON/QUANT/Q10,Q25,Q50,Q100,Q200,Q500
      COMMON/LP3NUM/R(100100)
      COMMON/MMPAR/A,B,C
      COMMON/SIZE/N,FN,DFN
      COMMON/NUMS/N1,N1NEG,N2,N2NEG,I1MIN,I1NMIN
      DIMENSION ISSIZE(10)
C=================================================================
C       NCSE = NO. OF CASES OF SAMPLE SIZES TO BE ANALYSED
C        M   = NO. OF MONTE-CARLO SAMPLES FOR EACH NCSE
C        N   = SIZE OF EACH SAMPLE (=ISSIZE(.))
C   AP,BP,CP = POPULATION PARAMETERS OF LP3 DISTRIBUTION
C      NRAN  = TOTAL NO. OF MONTE-CARLO NUMBERS GENERATED FOR GIVEN
C              POPULATION PARAMETERS (M*N .LESS THAN OR EQUAL TO. NRAN)
C=================================================================
C
      NCSE=1
C
      ISSIZE(1)=75
C     ISSIZE(2)=50
C     ISSIZE(3)=30
C     ISSIZE(4)=50
C     ISSIZE(5)=75
C
      M=1000
      NRAN=75000
C
      DSEED=123457.D0
      AP=0.059798
      BP=98.38009
      CP=-6.066213
      COVP=0.7
      SKP= 3.0
C
      WRITE(6,*)' CASE   5   --   C.V. = 0.7,   SKEW = 3 '
      WRITE(6,*)'          '
      WRITE(6,*)' POPULATION PARM. : ',AP,BP,CP,'| CV, SKEW :',COVP,SKP
      WRITE(10,*)AP,BP,CP,COVP,SKP,DSEED
C     ----------------------------------
```

```
      CALL LP3GN4(AP,BP,CP,DSEED,NRAN)
C     ---------------------------------
      DO 10 I=1,NCSE                                                    30
      N=ISSIZE(I)
      FN=FLOAT(N)
      DFN=FN
      WRITE(6,*)' SAMPLE SIZE = ',N,' NO. OF SAMPLES = ',M
      WRITE(10,*)N,M
C
      N1=0
      N1NEG=0
      N2=0
      N2NEG=0
      I1MIN=0
      I1NMIN=0
C
      DO 20 J=1,M
      KL=N*(J-1)+1
      KU=N*J
C
      CALL STDIZE(KL,KU,XM)
      CALL SERCH1(CP,C,A,B)
C
C     CALL SCAN(CP,C1,C2,C2NEG,FLLD1,FLLD2,FLLD2N)
C     CALL SEARCH(CP,C1,C2,C2NEG,FLLD1,FLLD2,FLLD2N)
C
C     WRITE(6,*)'|MAIN| CP = ',CP
C     WRITE(6,*)'|MAIN| C1 = ',C1,FLLD1
C     WRITE(6,*)'|MAIN| C2 = ',C2,FLLD2
C     WRITE(6,*)'|MAIN| C2NEG = ',C2NEG,FLLD2N
C
C     CALL PARMS(C1,C2,C2NEG,FLLD1,FLLD2,FLLD2N,C,A,B)
C
C     WRITE(6,*)' |PARMS|  C, A, B =  ',C,A,B
C
C-----------------------------------------------------
      SKX=SKP
      XMR=XM
      CPMY=C+A*B
      VARL=B*A*A
      STDL=SQRT(VARL)
      SKL=2.*(ABS(A)/A)*(1./SQRT(B))
      IF(ABS(SKL).LE.5.5)GO TO 30
      WRITE(6,*)' LOG SKEW = ',SKL
      GO TO 40
  30  CALL LPQNTL(CPMY,STDL,SKL,XMR)
C
  40  WRITE(6,11)J,A,B,C,SKX,Q10,Q25,Q50,Q100,Q200,Q500
      WRITE(10,11)J,A,B,C,SKX,Q10,Q25,Q50,Q100,Q200,Q500
  11  FORMAT(1X,I4,F16.7,E15.7,F17.7,F13.7,6F10.5)
C
  20  CONTINUE
      WRITE(6,*)' NO. OF C1 (+VE) = ',N1
      WRITE(6,*)' NO. OF C1 (-VE) = ',N1NEG
      WRITE(6,*)' NO. OF C2 (+VE) = ',N2
      WRITE(6,*)' NO. OF C2 (-VE) = ',N2NEG
      WRITE(6,*)' NO. OF MINIMA IN C1 (+VE) REGION = ',I1MIN
      WRITE(6,*)' NO. OF MINIMA IN C1 (-VE) REGION = ',I1NMIN
      WRITE(6,*)'                '
      WRITE(6,*)'=================================================
```

```
      @=============================================================='
   10 CONTINUE
      STOP
      END
C
C               ***   END OF MAIN SEGMENT   ***
C
C-------------------------------------------------------------------
C     SUBROUTINE LP3GN4 GENERATES LP3 NUMBERS USING IMSL ROUTINE
C               GGAMR (GAMMA GENERATOR).
C-------------------------------------------------------------------
      SUBROUTINE LP3GN4(A,B,C,DSEED,NR)
C
      DOUBLE PRECISION DSEED
      DOUBLE PRECISION DT,DR,DDR
      COMMON/LP3NUM/R(100100)
      DIMENSION WK(1)
C
      CALL GGAMR(DSEED,B,NR,WK,R)
C
      DO 10 I=1,NR
      T=A*R(I)+C
      DT=T
      DR=DEXP(DT)
      R(I)=DR
   10 CONTINUE
C
      RETURN
      END
C
C-------------------------------------------------------------------
C    STDIZE : STANDARDIZES AN LP SAMPLE, SORTS IT, AND TRANSORMS THE
C          SORTED SAMPLE THROUGH NATURAL LOG ( X(.) TO XK(.) TO YK(.) ).
C-------------------------------------------------------------------
      SUBROUTINE STDIZE(KL,KU,XM)
C
      DOUBLE PRECISION YK,DXK,DFN
      DIMENSION XK(100)
      COMMON/LP3NUM/X(100100)
      COMMON/P3YK/YK(100)
      COMMON/SIZE/N,FN,DFN
      S=0.
      DO 10 I=KL,KU
   10 S=S+X(I)
      XM=S/FN
      DO 11 I=KL,KU
   11 XK(I-KL+1)=X(I)/XM
C
      CALL VSRTA(XK,N)
C
      DO 12 I=1,N
      DXK=XK(I)
   12 YK(I)=DLOG(DXK)
C
C     CALL UBVSK(XK,XKM,XKV,XKSK)
C     WRITE(6,*)'SAMPLE STATS ='
C     WRITE(6,*)'  OBSERVED    ',XKM,XKV,XKSK
      RETURN
      END
C
```

```
C
C-----------------------------------------------------------------------
C   SERCH1 : SUBROUTINE TO FIND THE 'BEST' MLE ROOT BASED ON THE LKHD FN
C-----------------------------------------------------------------------
      SUBROUTINE SERCH1(CP,C,A,B)
C
      DOUBLE PRECISION YK,CLOW,CUP,CSTART,CDEL,DFN
      DOUBLE PRECISION DELFLL,DC,DA,DB,DR
      COMMON/SIZE/N,FN,DFN
      COMMON/NUMS/N1,N1NEG,N2,N2NEG,I1MIN,I1NMIN
      COMMON/P3YK/YK(100)
C
      IF(CP.GT.0.0)THEN
C-----------------------------------------SEARCH FOR C1 (ABOVE YMAX)
      CLOW=YK(N)+0.01D0
      CUP=YK(N)+50.0D0
      NINTC=50
      CALL CSMALL(CLOW,CUP,NINTC,C1,DELFLL,IY)
          IF (IY.GT.0) THEN
                IF(DELFLL.GE.0.0D0) THEN
                   C=C1
                   N1=N1+1
                   GO TO 10
                ELSE
                   CSTART=YK(N)+50.0D0
                   CDEL=50.0D0
                   CALL CLARGE(CSTART,CDEL,C2)
                   C=C2
                   N2=N2+1
                   I1MIN=I1MIN+1
                   GO TO 10
                END IF
            ELSE
                CLOW=YK(1)-0.01D0
                CUP=YK(1)-50.0D0
                NINTC=50
                CALL CSMALL(CLOW,CUP,NINTC,C1,DELFLL,IY)
                IF (IY.GT.0) THEN
                      IF(DELFLL.GE.0.0D0) THEN
                         C=C1
                         N1NEG=N1NEG+1
                         GO TO 10
                      ELSE
                         CSTART=YK(N)+50.0D0
                         CDEL=50.0D0
                         CALL CLARGE(CSTART,CDEL,C2)
                         C=C2
                         N2=N2+1
                         I1NMIN=I1NMIN+1
                         GO TO 10
                      END IF
                   ELSE
                      CSTART=YK(N)+50.0D0
                      CDEL=50.0D0
                      CALL CLARGE(CSTART,CDEL,C2)
                      C=C2
                      N2=N2+1
                      GO TO 10
                   END IF
               END IF
```

```
C     ==========================================
      ELSE
C     ==========================================
C-----------------------------------------SEARCH FOR C1 (BELOW YMIN)
      CLOW=YK(1)-0.01D0
      CUP=YK(1)-50.0D0
      NINTC=50
      CALL CSMALL(CLOW,CUP,NINTC,C1,DELFLL,IY)
          IF (IY.GT.0) THEN
                IF(DELFLL.GE.0.0D0) THEN
                   C=C1
                   N1NEG=N1NEG+1
                   GO TO 10
                ELSE
                   CSTART=YK(1)-50.0D0
                   CDEL=-50.0D0
                   CALL CLARGE(CSTART,CDEL,C2)
                   C=C2
                   N2NEG=N2NEG+1
                   I1NMIN=I1NMIN+1
                   GO TO 10
                END IF
            ELSE
                 CLOW=YK(N)+0.01D0
                 CUP=YK(N)+50.0D0
                 NINTC=50
                 CALL CSMALL(CLOW,CUP,NINTC,C1,DELFLL,IY)
                 IF (IY.GT.0) THEN
                       IF(DELFLL.GE.0.0D0) THEN
                          C=C1
                          N1=N1+1
                          GO TO 10
                       ELSE
                          CSTART=YK(1)-50.0D0
                          CDEL=-50.0D0
                          CALL CLARGE(CSTART,CDEL,C2)
                          C=C2
                          N2NEG=N2NEG+1
                          I1MIN=I1MIN+1
                          GO TO 10
                       END IF
                 ELSE
                          CSTART=YK(1)-50.0D0
                          CDEL=-50.0D0
                          CALL CLARGE(CSTART,CDEL,C2)
                          C=C2
                          N2NEG=N2NEG+1
                          GO TO 10
                 END IF
            END IF
      END IF
C
 10   DC=C
      CALL RVALUE (DC,DA,DB,DR)
      A=DA
      B=DB
      RETURN
      END
C
C-----------------------------------------------------------------------
```

```
C    CSMALL  :   SUBROUTINE TO SEARCH AND FIND THE ROOT 'C1'(IF EXISTS)
C                IN THE LOWER RANGE
C-----------------------------------------------------------------------
      SUBROUTINE CSMALL(CLOW,CUP,NINTC,C1,DELFLL,IY)
C
      DOUBLE PRECISION C,CNXT,A,B,R,RNXT,PROD,DDELC,CLOW,CUP,DELFLL
      DDELC=(CUP-CLOW)/DFLOAT(NINTC)
      IY=0
      C1=0.0
      C=CLOW
      CALL RVALUE(C,A,B,R)
C     CALL FNLKD(A,B,C,FLLD1)
C     WRITE(6,*)'|CSMALL|   C = ',C,' R = ',R,' LKHD FN. = ',FLLD1
      CNXT=C
      DO 10  I=1,NINTC
      CNXT=CNXT+DDELC
      CALL RVALUE(CNXT,A,B,RNXT)
C     CALL FNLKD(A,B,CNXT,FLLD1)
C     WRITE(6,*)'|CSMALL|   C = ',CNXT,' R = ',RNXT,' LKHD FN. = ',FLLD1
      PROD=R*RNXT
      IF(PROD.LE.0.0D0)GO TO 20
      R=RNXT
      C=CNXT
 10   CONTINUE
      GO TO 30
 20   CALL BISECT(C,R,CNXT,RNXT,C1,DELFLL)
      IY=1
 30   RETURN
      END
C
C-----------------------------------------------------------------------
C    BISECT : SUBROUTINE TO FIND THE ROOT 'XROOT' IN THE INTERVAL(X1,X2)
C             IN WHICH F(X) CHANGES SIGN.
C-----------------------------------------------------------------------
      SUBROUTINE BISECT(X1,F1,X2,F2,XROOT,DELFLL)
C
      DOUBLE PRECISION X1,F1,X2,F2,XNUM,XDENOM,X3,F3,A,B,PROD
      DOUBLE PRECISION FLLD1,FLLD3,DELFLL
C
      CALL RVALUE(X1,A,B,F1)
      CALL FNLKD(A,B,X1,FLLD1)
C
 20   XNUM=F2*X1-F1*X2
      XDENOM=F2-F1
      X3=XNUM/XDENOM
      CALL RVALUE(X3,A,B,F3)
C
      IF(DABS(F3).LE.1.0D-06)GO TO 10
      PROD=F1*F3
          IF(PROD.GE.0.0D0)THEN
          X1=X3
          F1=F3
          GO TO 20
          ELSE
          X2=X3
          F2=F3
          GOTO 20
          END IF
 10   XROOT=X3
      CALL FNLKD(A,B,X3,FLLD3)
```

```
      DELFLL=FLLD3-FLLD1
C
C     AA=A
C     BB=B
C     CALL ESTLP3(AA,BB,XROOT,XKM,XKV)
C     WRITE(6,*)'  MLE EST.     ',XKM,XKV
      RETURN
      END
C
C-----------------------------------------------------------------------
C    CLARGE : SUBROUTINE TO FIND THE ASYMPTOTIC ROOT 'C2' IN LARGE RANGE
C-----------------------------------------------------------------------
      SUBROUTINE CLARGE(CSTART,CDEL,C2)
C
      DOUBLE PRECISION C,CDEL,A,B,R,CSTART
      C=CSTART
 20   CALL RVALUE(C,A,B,R)
      IF(DABS(R).LE.1.0D-9)GO TO 10
      C=C+CDEL
      GO TO 20
 10   C2=C
C
C     AA=A
C     BB=B
C     CALL ESTLP3(AA,BB,C2,XKM,XKV)
C     WRITE(6,*)'  MLE EST.     ',XKM,XKV
      RETURN
      END
C
C-----------------------------------------------------------------------
C     RVALUE : SUBROUTINE TO COMPUTE PARAMETERS A, B AND RESIDUAL R IN
C              MLE EQUATIONS FOR A SPECIFIED LOCATION PARAMETER 'DC'
C-----------------------------------------------------------------------
      SUBROUTINE RVALUE(DC,DA,DB,DR)
C
      DOUBLE PRECISION MMPSI,DA,DB,DC,PSIB
      DOUBLE PRECISION YK,DFN
      DOUBLE PRECISION DS1,DS2,DS3,DS1S2
      DOUBLE PRECISION DT,DTINV,DTT,DR
      COMMON/P3YK/YK(100)
      COMMON/SIZE/N,FN,DFN
      COMMON/SUMM/DS1,DS3
      DS1=0.D0
      DS2=0.D0
      DO 10 I=1,N
      DT=YK(I)-DC
      DTINV=1.0D0/DT
      DS1=DS1+DT
 10   DS2=DS2+DTINV
C
      DS1S2=DS1*DS2
      DB=DS1S2/(DS1S2-DFN*DFN)
      DA=DS1/(DFN*DB)
      PSIB=MMPSI(DB,IER)
C
      DS3=0.D0
      DO 11 I=1,N
      DTT=DLOG((YK(I)-DC)/DA)
 11   DS3=DS3+DTT
      DR=-DFN*PSIB+DS3
```

```
      RETURN
      END
C
C-----------------------------------------------------------------------
C    FNLKD: SUBROUTINE TO COMPUTE THE LOG-LIKELIHOOD FUNCTION.
C-----------------------------------------------------------------------
      SUBROUTINE FNLKD(DA,DB,DC,FLLD)
C
      DOUBLE PRECISION DA,DB,DC,DS1,DS3,DFN,YK,DLGAMA,DLGAM
      DOUBLE PRECISION DT1,DT2,DT3,FLLD,DGMB
      COMMON/SUMM/DS1,DS3
      COMMON/SIZE/N,FN,DFN
      DLGAM=DLGAMA(DB)
      DT1=DLOG(DABS(DA))+DC+DLGAM
      DT1=DFN*DT1
      DT2=DS1*(1.D0+1.D0/DA)
      DT3=(DB-1.D0)*DS3
      FLLD=-DT1-DT2+DT3
      RETURN
      END
C
C
C**************************************************************************
C--LPQNTL - SUB-ROUTINE TO COMPUTE LOG PEARSON QUANTILES. PEARSON
C  FACTORS GIVEN IN WRC BULLETIN #17 (K-TABLES) ARE LINEARLY
C  INTERPOLATED
C**************************************************************************
      SUBROUTINE LPQNTL(XM,STD,SK,XMR)
C
      COMMON/QUANT/Q10,Q25,Q50,Q100,Q200,Q500
      DIMENSION XK(15,111),CDF(15),RTPK(15),RTLF(15),K(15),Q(15),
     1XJ(111),X1(111),X2(111),X3(111),X4(111),X5(111),X6(111),
     1X7(111),X8(111),X9(111),X10(111),X11(111),X12(111),
     1X13(111),X14(111),X15(111)
      REAL K
      DATA CDF/.005,.01,.02,.04,.1,.2,.5,.8,.9,.96,.98,.99,.995,.998,
     1.999/
      DATA X1/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.606,-.625,-.6452,
     3-.6667,-.6896,-.7143,-.7407,-.7691,-.7997,-.8328,-.8686,-.9074,
     4-.9495,-.995,-1.0443,-1.0975,-1.1548,-1.2162,-1.2817,-1.3511,
     5-1.4244,-1.5011,-1.5811,-1.6639,-1.7492,-1.8366,-1.9258,-2.0164,
     6-2.10825,-2.2009,-2.2942,-2.388,-2.4819,-2.5758,
     70.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,
     80.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,25*0./
      DATA X2/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.6061,-.625,-.6451,
     1-0.6666,-0.6896,-.7145,-0.7405,-0.7688,-0.7992,-0.832,-0.8672,
     1-0.9052,-0.9461,-0.99,-1.037,-1.0871,-1.1404,-1.1968,-1.2561,
     2-1.3182,-1.3827,-1.4494,-1.5181,-1.5884,-1.66,-1.7327,-1.8062,
     3-1.8803,-1.9547,-2.0293,-2.1039,-2.1784,-2.2526,-2.3264,55*0./
      DATA X3/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.6061,-.625,-.6451,
     1-0.6665,-0.6894,-.7138,-0.7399,-0.7678,-0.79765,-0.8296,-0.8637,
     1-0.9001,-0.9388,-0.9798,-1.0231,-1.0686,-1.1163,-1.1658,-1.2172,
     1-1.27,-1.3241,-1.3793,-1.4353,-1.4919,-1.5489,-1.606,-1.6633,
     2-1.7203,-1.7772,-1.8336,-1.8896,-1.945,-1.9997,-2.0538,55*0./
```

```
 DATA X5/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
1-.4255,-.4348,-.44443,-.45452,-.4651,-.4761,-.4877,-.4999,
2-.5126,-.526,-.5401,-.5548,-.57035,-.5867,-.6038,-.62175,-.6406,
1-0.6602,-0.6808,-.7021,-0.7242,-0.7471,-0.7706,-0.7947,
1-0.8193,-0.8442,-0.8694,-0.8946,-0.9199,-0.945,-0.9698,-0.9942,
2-1.0181,-1.0414,-1.0641,-1.0861,-1.1073,-1.1276,-1.1471,-1.1657,
3-1.1835,-1.2003,-1.2162,-1.2311,-1.2452,-1.2582,-1.2704,-1.2816,
455*0./
 DATA X6/-.3636,-.3704,-.37734,-.38458,-.39211,-.39993,-.40806,
1-.4165,-.4253,-.4345,-.444,-.454,-.4643,-.475,-.4862,-.4978,
2-.5099,-.5224,-.5353,-.5487,-.5624,-.5765,-.591,-.6057,-.6206,
1-0.6357,-0.6509,-.666,-0.6811,-0.696,-0.7107,-0.725,-0.7388,
1-0.7521,-0.7648,-0.7769,-0.7882,-0.7987,-0.8084,-0.8172,-0.8252,
2-.8322,-.8384,-.8437,-.8481,-.8516,-.8543,-.8561,-.857,-.8572,
3-.8565,-.8551,-.8529,-.8499,-.8461,-.8416,55*0./
 DATA X7/-.3546,-.3596,-.3645,-.3695,-.3743,-.379,-.3836,-.388,
1-.3922,-.3962,-.3999,-.4032,-.4062,-.4088,-.411,-.4127,-.4138,
2-.4144,-.4144,-.4138,-.4125,-.4106,-.4079,-.4045,-.4004,
1-.3955,-.3899,-.3835,-.3764,-.3685,-.3599,-.3506,-.3406,
1-.33,-.3187,-.3069,-.2944,-.2815,-.2681,-.2542,-.24,-.2254,
2-.2104,-.1952,-.1797,-.164,-.1481,-.132,-.1158,-.0995,-.083,
3-.0665,-.0499,-.0333,-.0166,0.,55*0./
 DATA X8/-.0103,.00243,.0156,.0293,.0434,.058,.073,.0885,.1044,
1.1207,.1374,.1545,.1719,.1897,.2078,.2262,.2448,.2638,.2829,.3022,
2.3217,.3413,.361,.3808,.4006,
1.4204,.4402,.4598,.4793,.4987,.5179,.5368,.5555,.5738,
1.5918,.6094,.6266,.6434,.6596,.6753,.6905,.7051,.7192,.7326,
2.7454,.7575,.769,.7799,.79,.7995,.8083,.8164,.8238,.8304,.8364,
3.8416,55*0./
 DATA X9/.6912,.712,.7328,.7536,.7746,.7955,.8164,.8373,.8582,
1.879,.8996,.9202,.9406,.9609,.981,1.0008,1.0204,1.0397,1.0586,
21.0773,1.0955,1.1134,1.1308,1.1477,1.1642,1.1801,1.1954,
11.2101,1.2242,1.2377,1.2504,1.2624,1.2737,1.2841,1.2938,
11.3026,1.3105,1.3176,1.3238,1.329,1.3333,1.3367,1.339,1.3405,
21.3409,1.3404,1.3389,1.3364,1.3329,1.3285,1.3231,1.3167,1.3094,
31.3011,1.2918,1.2816,55*0./
 DATA X10/2.0474,2.0637,2.0795,2.0949,2.1099,2.1243,2.1383,2.1517,
12.1647,2.177,2.1887,2.1999,2.2104,2.2202,2.2294,2.2379,2.2456,
22.2525,2.2587,2.2641,2.2686,2.2723,2.2751,2.2769,2.2779,2.2778,
12.27676,2.2747,2.2716,2.2674,2.2622,2.2558,2.2483,2.2397,
12.2299,2.2189,2.2067,2.1933,2.1787,2.1629,2.1459,2.1277,2.1082,
22.0876,2.0657,2.0427,2.0185,1.9931,1.9666,1.939,1.9102,1.8804,
31.8495,1.8176,1.7846,1.7507,1.7158,1.68,1.6433,1.6057,1.5674,
41.5283,1.4885,1.4481,1.4072,1.3658,1.3241,1.2823,1.2403,1.1984,
51.1568,1.1157,1.0751,1.0354,.9967,.9592,.923,.8881,.8549,.8232,
6.7931,.7646,.7377,.7123,.6884,.6659,
6.6447,.6247,.6059,.5881,.5714,.5555,.5405,.5263,.5128,.5,.4878,
7.4762,.4651,.4546,.4444,.4348,.4255,.4167,.4082,.4,.3922,.3846,
8.3774,.3704,.3636/
 DATA X11/3.2838,3.2884,3.2924,3.2957,3.2982,3.30007,3.3012,
13.3015,3.301,3.2998,3.2977,3.2947,3.2909,3.2862,3.2806,3.274,
23.2665,3.258,3.2485,3.238,3.2264,3.2138,3.2,3.1851,3.1691,
33.1519,3.1336,3.114,3.0932,3.0712,3.0479,3.0233,2.9974,2.9703,
42.9418,2.912,2.8809,2.8485,2.8147,2.7796,2.7433,2.7056,2.6666,
52.6263,2.5848,2.5421,2.4981,2.453,2.4067,2.3593,2.3108,2.2613,
62.2108,2.1594,2.107,2.0538,55*0./
 DATA X12/4.6402,4.6285,4.6159,4.6025,4.5882,4.573,4.5569,4.5399,
14.5219,4.503,4.483,4.4621,4.4401,4.4171,4.393,4.3678,4.3415,4.314,
24.2855,4.2557,4.2247,4.1926,4.1592,4.1245,4.0886,
34.0514,4.0129,3.973,3.9318,3.8893,3.8454,3.8001,3.7535,3.7054,
```

```
     43.656,3.6052,3.553,3.4994,3.4444,3.388,3.3304,3.2713,3.211,
     53.1494,3.0866,3.0226,2.9574,2.891,2.8236,2.7551,2.6857,2.6154,
     62.5442,2.4723,2.3996,2.3264,55*0./
      DATA X13/6.08307,6.0517,6.0193,5.986,5.9517,5.9164,5.88,5.8427,
     15.8042,5.7646,5.724,5.6822,5.6393,5.5953,5.5501,5.5036,5.456,
     25.4071,5.357,5.3056,5.2529,5.1989,5.1436,5.087,5.029,
     34.9696,4.9088,4.8467,4.7831,4.7182,4.6518,4.5839,4.5147,4.444,
     44.3719,4.2983,4.2234,4.147,4.0693,3.9902,3.9097,3.828,3.745,
     53.6607,3.5753,3.4887,3.4011,3.3124,3.2228,3.1323,3.041,2.949,
     62.8564,2.7632,2.6697,2.5758,55*0./
      DATA X14/8.0869,8.0259,7.9639,7.9008,7.8366,7.7712,7.7048,
     17.6372,7.5684,7.4985,7.4273,7.355,7.2814,7.2065,7.1304,7.053,
     26.9744,6.8944,6.813,6.7303,6.6463,6.5608,6.474,6.3858,6.2961,
     16.20506,6.1125,6.0186,5.9232,5.8263,5.728,5.6282,5.5269,
     15.4243,5.3201,5.2146,5.1077,4.9994,4.8897,4.7788,4.6665,4.553,
     24.4384,4.3226,4.2058,4.088,3.9693,3.8498,3.7296,3.6087,3.4874,
     33.3657,3.2437,3.1217,2.9998,2.8782,2.7571,2.6367,2.5174,
     42.3994,2.2831,2.1688,2.057,1.9481,1.8424,1.7406,1.6431,1.5502,
     51.4623,1.3798,1.3028,1.2313,1.1653,1.1047,1.049,.998,.9513,.9085,
     6.8693,.8332,.7999,.7692,.7407,.7143,.6896,.6667,25*0./
      DATA X15/9.6577,9.5723,9.4859,9.3983,9.3095,9.2196,9.1285,9.0362,
     18.9427,8.848,8.752,8.6548,8.5563,8.4565,8.3553,8.2529,8.1491,
     28.044,7.9374,7.8295,7.7202,7.6095,7.4974,7.3838,7.2688,
     17.1524,7.0344,6.9151,6.7942,6.6719,6.5481,6.4229,6.2963,
     26.1682,6.0387,5.9078,5.7755,5.6419,5.507,5.3709,5.2335,5.0951,
     34.9555,4.8149,4.6734,4.5311,4.3881,4.2444,4.1002,3.9557,3.8109,
     43.6661,3.5214,3.377,3.2332,3.0902,2.9483,2.8079,2.6692,2.5326,
     52.3987,2.2678,2.1405,2.0174,1.8989,1.7857,1.6783,1.577,1.4822,
     61.3941,1.3128,1.2381,1.1697,1.1074,1.0507,.999,.9519,.9089,.8695,
     7.8333,.8,.7692,.7407,.7143,.6897,.6667,25*0./
      J=111
      DO 61 I=1,25
      X14(J)=-X1(I)
      X15(J)=-X1(I)
61    J=J-1
      J=111
      DO 62 I=1,111
      X4(J)=-X10(I)
62    J=J-1
      J=111
      DO 63 I=1,55
      X1(J)=-X13(I)
      X2(J)=-X12(I)
      X3(J)=-X11(I)
      X5(J)=-X9(I)
      X6(J)=-X8(I)
      X7(J)=-X7(I)
      X8(J)=-X6(I)
      X9(J)=-X5(I)
      X11(J)=-X3(I)
      X12(J)=-X2(I)
      X13(J)=-X1(I)
63    J=J-1
      DO 1 J =1,111
      XK(1,J)=X1(J)
      XK(2,J)=X2(J)
      XK(3,J)=X3(J)
      XK(4,J)=X4(J)
      XK(5,J)=X5(J)
      XK(6,J)=X6(J)
```

```
      XK(7,J)=X7(J)
      XK(8,J)=X8(J)
      XK(9,J)=X9(J)
      XK(10,J)=X10(J)
      XK(11,J)=X11(J)
      XK(12,J)=X12(J)
      XK(13,J)=X13(J)
      XK(14,J)=X14(J)
      XK(15,J)=X15(J)
    1 CONTINUE
      DO 65 I=1,15
      RTLF(I)=1./CDF(I)
   65 RTPK(I)=1./(1.-CDF(I))
      RTPK(15)=1000.
      RTPK(14)=500.
      RTPK(13)=200.
      RTPK(12)=100.
      J=1
  301 W=J
      XJ(J)=5.6-W/10.0
      IF(XJ(J)-SK)303,303,302
  302 J=J+1
      GO TO 301
  303 DO 304 I=9,14
      VK =((SK-XJ(J))*(XK(I,J-1)-XK(I,J)))/(XJ(J-1)-XJ(J))+XK(I,J)
      K(I)=EXP(XM+VK*STD)
  304 CONTINUE
      DO 305 I=9,14
  305 Q(I)=K(I)*XMR
      Q10=Q(9)
      Q25=Q(10)
      Q50=Q(11)
      Q100=Q(12)
      Q200=Q(13)
      Q500=Q(14)
C
C     WRITE(6,310)
C310  FORMAT(/' *** LOG PEARSON VARIATE ESTIMATES BY MXM1 METHOD ***'//
C    19X,'CDF',2X,'T(FOR LOS)',2X,'T(FOR PKS)',5X,'VARIATE',//)
C     DO 315 I=1,15
C     WRITE(6,320) CDF(I),RTLF(I),RTPK(I),Q(I)
C315  CONTINUE
C     WRITE(18,330)Q(7),Q(8),Q(9),Q(11),Q(12),Q(13),Q(14)
C330  FORMAT(3X,'MOMIX',1X,7(2X,F7.1))
C320  FORMAT(3F12.3,F12.2)
C     WRITE(6,325)
C325  FORMAT(' NOTE: T=RETURN PERIOD(YRS), LOS=MINIMUM VALUES LIKE LOW F
C    1LOWS, PKS=MAXIMUM VALUES LIKE FLOOD FLOWS')
      RETURN
      END
C
C
C-----------------------------------------------------------------------
C     SEARCH : SUBROUTINE TO SEARCH FOR THE ROOT OF 'C' IN BOTH +VE
C              AND -VE RANGE OF SMALL AS WELL AS LARGE VALUES.
C
      SUBROUTINE SEARCH(CP,C1,C2,C2NEG,FLLD1,FLLD2,FLLD2N)
      DOUBLE PRECISION YK,CLOW,CUP,CSTART,CDEL,DFN
      DOUBLE PRECISION FLLD1,FLLD2,FLLD2N
      COMMON/SIZE/N,FN,DFN
```

```
      COMMON/P3YK/YK(100)
      C1=0.0
      C2=0.0
      C2NEG=0.0
      FLLD1=-1000.0D0
      FLLD2=-1000.0D0
      FLLD2N=-1000.0D0
      IF(CP.GT.0.0)THEN
C-----------------------------------------SEARCH FOR C1 (ABOVE YMAX)
      CLOW=YK(N)+0.01D0
      CUP=YK(N)+50.0D0
      NINTC=50
      CALL CSMALL(CLOW,CUP,NINTC,C1,FLLD1)
           IF (C1.NE.0.0) THEN
C-------------------------------------------------------------FIND C2NEG
           CSTART=YK(1)-50.0D0
           CDEL=-50.0D0
           CALL CLARGE(CSTART,CDEL,C2NEG,FLLD2N)
           RETURN
           ELSE
C---------------------------------------------------------------FIND C2
           CSTART=YK(N)+50.0D0
           CDEL=50.0D0
           CALL CLARGE(CSTART,CDEL,C2,FLLD2)
C-----------------------------------------SEARCH FOR C1 (BELOW YMIN)
           CLOW=YK(1)-0.01D0
           CUP=YK(1)-50.0D0
           NINTC=50
           CALL CSMALL(CLOW,CUP,NINTC,C1,FLLD1)
                IF (C1.NE.0.0) THEN
                RETURN
                ELSE
C-------------------------------------------------------------FIND C2NEG
                CSTART=YK(1)-50.0D0
                CDEL=-50.0D0
                CALL CLARGE(CSTART,CDEL,C2NEG,FLLD2N)
                RETURN
                END IF
           END IF
C     =====
      ELSE
C     =====
C-----------------------------------------SEARCH FOR C1 (BELOW YMIN)
      CLOW=YK(1)-0.01D0
      CUP=YK(1)-50.0D0
      NINTC=50
      CALL CSMALL(CLOW,CUP,NINTC,C1,FLLD1)
           IF (C1.NE.0.0) THEN
C---------------------------------------------------------------FIND C2
           CSTART=YK(N)+50.0D0
           CDEL=50.0D0
           CALL CLARGE(CSTART,CDEL,C2,FLLD2)
           RETURN
           ELSE
C-------------------------------------------------------------FIND C2NEG
           CSTART=YK(1)-50.0D0
           CDEL=-50.0D0
           CALL CLARGE(CSTART,CDEL,C2NEG,FLLD2N)
C-----------------------------------------SEARCH FOR C1 (ABOVE YMAX)
           CLOW=YK(N)+0.01D0
```

```
            CUP=YK(N)+50.0D0
            NINTC=50
            CALL CSMALL(CLOW,CUP,NINTC,C1,FLLD1)
                 IF (C1.NE.0.0) THEN
                 RETURN
                 ELSE
C-----------------------------------------------------------------FIND C2
                 CSTART=YK(N)+50.0D0
                 CDEL=50.0D0
                 CALL CLARGE(CSTART,CDEL,C2,FLLD2)
                 RETURN
                 END IF
            END IF
       END IF
       RETURN
       END
C
C-----------------------------------------------------------------------
C      SCAN : SUBROUTINE TO SCAN THE C DOMAIN FOR POSSIBLE ROOTS
C               AND TO STUDY THE BEHAVIOUR OF LKHD FN.
C
       SUBROUTINE SCAN(CP,C1,C2,C2NEG,FLLD1,FLLD2,FLLD2N)
       DOUBLE PRECISION YK,CLOW,CUP,CSTART,CDEL,DFN
       DOUBLE PRECISION FLLD1,FLLD2,FLLD2N
       COMMON/SIZE/N,FN,DFN
       COMMON/P3YK/YK(100)
C
       CLOW=YK(N)+0.01D0
       CUP=YK(N)+50.0D0
       NINTC=50
       CALL CSMALL(CLOW,CUP,NINTC,C1,FLLD1)
C
       IF(C1.NE.0.0)THEN
       CSTART=C1
       ELSE
       CSTART=CUP
       END IF
       CDEL=25.0D0
       CALL CLARGE(CSTART,CDEL,C2NEG,FLLD2N)
C
       CLOW=YK(1)-0.01D0
       CUP=YK(1)-50.0D0
       NINTC=50
       CALL CSMALL(CLOW,CUP,NINTC,C1,FLLD1)
C
       IF(C1.NE.0.0)THEN
       CSTART=C1
       ELSE
       CSTART=CUP
       END IF
       CDEL=-25.0D0
       CALL CLARGE(CSTART,CDEL,C2NEG,FLLD2N)
       RETURN
       END
C
C-----------------------------------------------------------------------
C   PARMS : PICKS THE LOCATION PARM. 'C' CORRESPONDING TO MAXIMUM VALUE
C           OF THE LOG-LIKELIHOOD FUNCTION. COMPUTES PARMS. A & B
C
       SUBROUTINE PARMS(C1,C2,C2NEG,FLLD1,FLLD2,FLLD2N,C,A,B)
```

```
      DOUBLE PRECISION DC,DA,DB,DR
      DOUBLE PRECISION FLLD1,FLLD2,FLLD2N
      IF(FLLD1.GE.FLLD2) GO TO 10
           IF(FLLD2.GE.FLLD2N) GO TO 20
                C=C2NEG
                DC=C
                CALL RVALUE(DC,DA,DB,DR)
                A=DA
                B=DB
                RETURN
C
  20            C=C2
                DC=C
                CALL RVALUE(DC,DA,DB,DR)
                A=DA
                B=DB
                RETURN
C
  10  IF(FLLD1.GE.FLLD2N) GO TO 30
                C=C2NEG
                DC=C
                CALL RVALUE(DC,DA,DB,DR)
                A=DA
                B=DB
                RETURN
C
  30            C=C1
                DC=C
                CALL RVALUE(DC,DA,DB,DR)
                A=DA
                B=DB
                RETURN
      END
C
C*************************************************************************
C     SERCH2 : SUBROUTINE TO SEARCH FOR TWO ROOTS OF THE LOCATION PARM.
C          'C' IN LOWER AND HIGHER REGION (REF : RAO'S MLE PAPER, 1986)
C*************************************************************************
      SUBROUTINE SERCH2(CSTART,CDEL,C1,C2)
      DOUBLE PRECISION DA,DB,DC,DR,CSTART,CDEL
      DOUBLE PRECISION FLIKE
      DC=CSTART
      ITR=1
C
 13   CALL RVALUE(DC,DA,DB,DR)
      CALL FNLKD(DA,DB,DC,FLIKE)
C
      WRITE(6,*)DC,DR,FLIKE
      DC=DC+CDEL
      IF(ITR.GE.25)GO TO 12
      ITR=ITR+1
      GO TO 13
C
 12   RETURN
      END
C
C
C*************************************************************************
C     QNTL : SUBROUTINE TO FIND THE QUANTILE BY CONVERTING THE LP3
C            VARIATE TO THE STANDARDIZED GAMMA VARIATE
```

```
C*************************************************************************
C
      SUBROUTINE QNTL(A,B,C,XM)
      DIMENSION RTP(6),EXPROB(6),Q(6)
      COMMON/QUANT/Q10,Q25,Q50,Q100,Q200,Q500
      DOUBLE PRECISION FNGAMA,B1
      RTP(1)=10.
      RTP(2)=25.
      RTP(3)=50.
      RTP(4)=100.
      RTP(5)=200.
      RTP(6)=500.
      DO 5 I=1,6
      EXPROB(I)=1./RTP(I)
 5    CONTINUE
      IF(A.GT.0.0) THEN
          DO 10 I=1,6
          XT=XM
          T=(ALOG(XT)-C)/A
30        CALL MDGAM(T,B,CUMF,IER)
          G=EXPROB(I)-(1.-CUMF)
          IF(G.LE.1.0E-06)GO TO 20
          FDENST=EXP(-T)*(T**(B-1.))/FNGAMA(B)
          T=T-G/FDENST
          GO TO 30
20        XT=EXP(A*T+C)
          Q(I)=XT*XM
10        CONTINUE
      ELSE
          DO 40 I=1,6
          XT=XM
          T=(ALOG(XT)-C)/A
60        CALL MDGAM(T,B,CUMF,IER)
          G=CUMF-EXPROB(I)
          IF(G.LE.1.0E-06)GO TO 50
          B1=B
          FDENST=EXP(-T)*(T**(B-1.))/FNGAMA(B1)
          T=T-G/FDENST
          GO TO 60
50        XT=EXP(A*T+C)
          Q(I)=XT*XM
40        CONTINUE
      END IF
      RETURN
      END
C
C     FNGAMA : USED BY THE SUBROUTINE 'QNTL' ABOVE
C
      FUNCTION FNGAMA(B)
      DOUBLE PRECISION B,B1,PROD,FNGAMA
      B1=B
      PROD=1.0D0
 10   B1=B1-1.D0
      PROD=PROD*B1
      IF(B1.LT.57.)THEN
          PROD=PROD*DGAMMA(B1)
          FNGAMA=PROD
          RETURN
      ELSE
          B1=B1-1.D0
```

```
          GO TO 10
        END IF
        RETURN
        END
C
C*************************************************************************
C     UBVSK: SUB-ROUTINE TO COMPUTE UNBIASED MEAN, VARIANCE, &
C     SKEWNESS COEFFICIENT (VM,VV,VSK RESPECTIVELY)
C*************************************************************************
C
        SUBROUTINE UBVSK(V,VM,VV,VSK)
        DOUBLE PRECISION DFN
        COMMON/SIZE/N,FN,DFN
        DIMENSION V(N)
C
        C1=FN/(FN-1.)
        C2=FN**2/(FN-1.)/(FN-2.)
        C2=C2/C1**1.5
C
        X1=0.
        X2=0.
        X3=0.
        DO 10 I=1,N
        V1=V(I)
        V2=V1*V1
        V3=V2*V1
C
        X1=X1+V1
        X2=X2+V2
 10     X3=X3+V3
C
        VM=X1/FN
        VV=X2/FN-VM**2
        VSK=(X3/FN-3.*VM*VV-VM**3)/VV**1.5
        VV=VV*C1
        VSK=VSK*C2
        RETURN
        END
C
C
        SUBROUTINE ESTLP3(A,B,C,XM,XV)
        DOUBLE PRECISION DA,DB,DC,DA1,DA2,DAL1,DAL2
        DOUBLE PRECISION XMLN,DXM,CV2LN1,CV2,DXV
        WRITE(6,*)' |ESTLP3|  A, B, C = ',A,B,C
        XM=0.0
        XV=0.0
        DA=A
        DB=B
        DC=C
        IF(DA.GE.1.D0) GO TO 10
        DA1=1.D0-DA
        DAL1=DLOG(DA1)
C
        XMLN=DC-DB*DAL1
        DXM=DEXP(XMLN)
        XM=DXM
C
        IF(DA.GE.0.5D0)GO TO 10
        DA2=1.D0-(2.D0*DA)
        DAL2=DLOG(DA2)
```

```
      CV2LN1=DB*(2.0D0*DAL1-DAL2)
      CV2=DEXP(CV2LN1)-1.0D0
      DXV=CV2*DXM*DXM
      XV=DXV
C
  10  RETURN
      END
$ENTRY
//GO.FT10F001 DD DSN=CEAROR.LP3.OUTPUTM8,DISP=SHR
$$
//
//GO.FT10F001 DD DSN=CEAROR.LP3.OUTPUTM7,DISP=SHR
```

```
//ENTROP5 JOB (1304,59634,50,20),'ARORA',MSGCLASS=S,CLASS=H              00010000
/*ROUTE PRINT CEBA                                                      00020000
//     EXEC WATFIV,LIB='CEAROR.SPEC.LIB',REGION.GO=4000K,TIME.GO=99
$JOB            TIME=4500,NOEXT
C
C=====================================================================
C                  LOG PEARSON TYPE 3 DISTRIBUTION
C---------------------------------------------------------------------
C     PROGRAM TO COMPUTE PARAMETERS AND QUANTILES BY THE METHOD OF
C     MAXIMUM ENTROPY (ENT).
C=====================================================================
C
C/*JOBPARM SHIFT=N
      DOUBLE PRECISION DSEED,DFN,YK,YKM,VYK
      DOUBLE PRECISION H1,H2,H2N
      COMMON/P3YK/YK(100),YKM,VYK
      COMMON/QUANT/Q10,Q25,Q50,Q100,Q200,Q500
      COMMON/LP3NUM/R(100100)
      COMMON/MMPAR/A,B,C
      COMMON/SIZE/N,FN,DFN
      COMMON/NUMS/N1,N2
      DIMENSION ISSIZE(10)
C=====================================================================
C       NCSE = NO. OF CASES OF SAMPLE SIZES TO BE ANALYSED
C        M   = NO. OF MONTE-CARLO SAMPLES FOR EACH NCSE
C        N   = SIZE OF EACH SAMPLE (=ISSIZE(.))
C   AP,BP,CP = POPULATION PARAMETERS OF LP3 DISTRIBUTION
C      NRAN  = TOTAL NO. OF MONTE-CARLO NUMBERS GENERATED FOR GIVEN
C              POPULATION PARAMETERS (M*N .LESS THAN OR EQUAL TO. NRAN)
C=====================================================================
      NCSE=5
C
      ISSIZE(1)=10
      ISSIZE(2)=20
      ISSIZE(3)=30
      ISSIZE(4)=50
      ISSIZE(5)=75
C
      M=1000
      NRAN=75000
      DSEED=123457.D0
      AP=0.059798
      BP=98.380090
      CP=-6.066213
      COVP=0.7
      SKP= 3.0
C
      WRITE(6,*)'           '
      WRITE(6,*)' CASE 5    :    C.V. = ',COVP,'  SKEW = ',SKP
      WRITE(6,*)'           '
      WRITE(6,*)' POPULATION PARM. : ',AP,BP,CP,'| CV, SKEW :',COVP,SKP
      WRITE(10,*)AP,BP,CP,COVP,SKP,DSEED
C     ---------------------------------
      CALL LP3GN4(AP,BP,CP,DSEED,NRAN)
```

```
C     ---------------------------------
      DO 10 I=1,NCSE
      N=ISSIZE(I)
      FN=FLOAT(N)
      DFN=FN
      WRITE(6,*)' SAMPLE SIZE = ',N,' NO. OF SAMPLES = ',M
      WRITE(10,*)N,M
C
      N1=0
      N2=0
C
      DO 20 J=1,M
C     WRITE(6,*)'=============================================================
C    @=============================================================='
C     WRITE(6,*)'          '
C     WRITE(6,*)' SAMPLE NO. = ',J
C     WRITE(6,*)'          '
      KL=N*(J-1)+1
      KU=N*J
C
      CALL STDIZE(KL,KU,XM)
      CALL SCAN1(CP,C,A,B)
C
C     WRITE(6,*)' ROOTS A, B, C = ',A,B,C
C
C     CALL SCAN(CP,C1,C2,C2NEG,H1,H2,H2N)
C     CALL SERCH1(CP,C,A,B)
C     CALL SEARCH(CP,C1,C2,C2NEG,FLLD1,FLLD2,FLLD2N)
C
C     WRITE(6,*)'|MAIN| CP = ',CP
C     WRITE(6,*)'|MAIN| C1 = ',C1,FLLD1
C     WRITE(6,*)'|MAIN| C2 = ',C2,FLLD2
C     WRITE(6,*)'|MAIN| C2NEG = ',C2NEG,FLLD2N
C
C     CALL PARMS(C1,C2,C2NEG,FLLD1,FLLD2,FLLD2N,C,A,B)
C
C     WRITE(6,*)' |PARMS|  C, A, B =  ',C,A,B
C
C-----------------------------------------------------
      SKX=SKP
      XMR=XM
      CPMY=C+A*B
      VARL=B*A*A
      STDL=SQRT(VARL)
      SKL=2.*(ABS(A)/A)*(1./SQRT(B))
      IF(ABS(SKL).LE.5.5)GO TO 30
      WRITE(6,*)' LOG SKEW = ',SKL
      GO TO 40
  30  CALL LPQNTL(CPMY,STDL,SKL,XMR)
C
  40  WRITE(6,11)J,A,B,C,SKX,Q10,Q25,Q50,Q100,Q200,Q500
      WRITE(10,11)J,A,B,C,SKX,Q10,Q25,Q50,Q100,Q200,Q500
  11  FORMAT(1X,I4,F16.7,E15.7,F17.7,F13.7,6F10.5)
C
  20  CONTINUE
C     WRITE(6,*)'          '
      WRITE(6,*)' NO. OF C1  = ',N1
      WRITE(6,*)' NO. OF C2  = ',N2
C     WRITE(6,*)'          '
C     WRITE(6,*)'=============================================================
```

```
C    @=============================================================='
  10  CONTINUE
      STOP
      END
C
C                ***   END OF MAIN SEGMENT   ***
C
C-----------------------------------------------------------------
C     SUBROUTINE LP3GN4 GENERATES LP3 NUMBERS USING IMSL ROUTINE
C                GGAMR (GAMMA GENERATOR).
C-----------------------------------------------------------------
      SUBROUTINE LP3GN4(A,B,C,DSEED,NR)
C
      DOUBLE PRECISION DSEED
      DOUBLE PRECISION DT,DR,DDR
      COMMON/LP3NUM/R(100100)
      DIMENSION WK(1)
C
      CALL GGAMR(DSEED,B,NR,WK,R)
C
      DO 10 I=1,NR
      T=A*R(I)+C
      DT=T
      DR=DEXP(DT)
      R(I)=DR
  10  CONTINUE
C
      RETURN
      END
C
C-----------------------------------------------------------------------
C    STDIZE : STANDARDIZES AN LP SAMPLE, SORTS IT, AND TRANSORMS THE
C          SORTED SAMPLE THROUGH NATURAL LOG ( X(.) TO XK(.) TO YK(.) ).
C-----------------------------------------------------------------------
      SUBROUTINE STDIZE(KL,KU,XM)
C
      DOUBLE PRECISION YK,DXK,DFN,DSS,DSS2,YKM,YKM2,VYK
      DIMENSION XK(100)
      COMMON/LP3NUM/X(100100)
      COMMON/P3YK/YK(100),YKM,VYK
      COMMON/SIZE/N,FN,DFN
      S=0.
      DO 10 I=KL,KU
 10   S=S+X(I)
      XM=S/FN
      DO 11 I=KL,KU
 11   XK(I-KL+1)=X(I)/XM
C
      CALL VSRTA(XK,N)
C
      DSS=0.0
      DSS2=0.0
      DO 12 I=1,N
      DXK=XK(I)
      YK(I)=DLOG(DXK)
      DSS=DSS+YK(I)
 12   DSS2=DSS2+YK(I)*YK(I)
      YKM=DSS/DFN
      YKM2=DSS2/DFN
      VYK=(YKM2-YKM*YKM)*(DFN/(DFN-1.0D0))
```

```
C     WRITE(6,*)'          '
C     WRITE(6,*)' |STDIZE|    YKM = ',YKM,' VYK = ',VYK
C     WRITE(6,*)'          '
C
      RETURN
      END
C
C-----------------------------------------------------------------------
C    SCAN1 : SUBROUTINE TO FIND THE ENT ROOT C IN THE DIRECTION OF CP
C              IN REGION 1 (SMALL VALUE), OR REGION 2 (LARGE VALUE)
C
      SUBROUTINE SCAN1(CP,C,A,B)
      DOUBLE PRECISION YK,CLOW,CUP,CSTART,CDEL,DFN,YKM,VYK
      DOUBLE PRECISION H1,H2,H2N,DELH1,CC,AA,BB,RR
      COMMON/SIZE/N,FN,DFN
      COMMON/P3YK/YK(100),YKM,VYK
      COMMON/NUMS/N1,N2
C
      IF (CP.GT.0.0) THEN
          CLOW=YK(N)+0.01D0
          CUP=YK(N)+50.0D0
          NINTC=50
          CALL CSMALL(CLOW,CUP,NINTC,C1,DELH1,IY)
C
               IF(C1.GT.0.D0)THEN
               CC=C1
               N1=N1+1
               GO TO 10
               ELSE
               CSTART=CUP
               CDEL=100.0D0
               CALL CLARGE(CSTART,CDEL,C2)
               CC=C2
               N2=N2+1
               GO TO 10
               END IF
      ELSE
C
          CLOW=YK(1)-0.01D0
          CUP=YK(1)-50.0D0
          NINTC=50
          CALL CSMALL(CLOW,CUP,NINTC,C1,DELH1,IY)
C
               IF(C1.LT.0.D0)THEN
               CC=C1
               N1=N1+1
               GO TO 10
               ELSE
               CSTART=CUP
               CDEL=-100.0D0
               CALL CLARGE(CSTART,CDEL,C2)
               CC=C2
               N2=N2+1
               GO TO 10
               END IF
C
      END IF
 10   CALL RVALUE(CC,AA,BB,RR)
      C=CC
      A=AA
```

```
      B=BB
      RETURN
      END
C
C-----------------------------------------------------------------
C     SCAN : SUBROUTINE TO SCAN THE C DOMAIN FOR POSSIBLE ROOTS
C              AND TO STUDY THE BEHAVIOUR OF ENTROPY FUNCTION
C
      SUBROUTINE SCAN(CP,C1,C2,C2NEG,H1,H2,H2N)
      DOUBLE PRECISION YK,CLOW,CUP,CSTART,CDEL,DFN,YKM,VYK
      DOUBLE PRECISION H1,H2,H2N,DELH1
      COMMON/SIZE/N,FN,DFN
      COMMON/P3YK/YK(100),YKM,VYK
C
      CLOW=YK(N)+0.01D0
      CUP=YK(N)+50.0D0
      NINTC=50
      CALL CSMALL(CLOW,CUP,NINTC,C1,DELH1,IY)
C
      IF(C1.NE.0.0)THEN
      CSTART=C1
      ELSE
      CSTART=CUP
      END IF
      CDEL=50.0D0
      CALL CLARGE(CSTART,CDEL,C2NEG)
C
      CLOW=YK(1)-0.01D0
      CUP=YK(1)-50.0D0
      NINTC=50
      CALL CSMALL(CLOW,CUP,NINTC,C1,DELH1,IY)
C
      IF(C1.NE.0.0)THEN
      CSTART=C1
      ELSE
      CSTART=CUP
      END IF
      CDEL=-50.0D0
      CALL CLARGE(CSTART,CDEL,C2NEG)
      RETURN
      END
C
C
C-----------------------------------------------------------------
C   SERCH1 : SUBROUTINE TO FIND THE 'BEST' MLE ROOT BASED ON THE LKHD FN
C-----------------------------------------------------------------
      SUBROUTINE SERCH1(CP,C,A,B)
C
      DOUBLE PRECISION YK,CLOW,CUP,CSTART,CDEL,DFN,YKM,VYK
      DOUBLE PRECISION DELFLL,DC,DA,DB,DR
      COMMON/SIZE/N,FN,DFN
C     COMMON/NUMS/N1,N1NEG,N2,N2NEG,I1MIN,I1NMIN
      COMMON/P3YK/YK(100),YKM,VYK
C
      IF(CP.GT.0.0)THEN
C-----------------------------------------SEARCH FOR C1 (ABOVE YMAX)
      CLOW=YK(N)+0.01D0
      CUP=YK(N)+50.0D0
      NINTC=50
      CALL CSMALL(CLOW,CUP,NINTC,C1,DELFLL,IY)
```

```
            IF (IY.GT.0) THEN
                  IF(DELFLL.GE.0.0D0) THEN
                     C=C1
                     N1=N1+1
                     GO TO 10
                  ELSE
                     CSTART=YK(N)+50.0D0
                     CDEL=50.0D0
                     CALL CLARGE(CSTART,CDEL,C2)
                     C=C2
                     N2=N2+1
                     I1MIN=I1MIN+1
                     GO TO 10
                  END IF
            ELSE
                CLOW=YK(1)-0.01D0
                CUP=YK(1)-50.0D0
                NINTC=50
                CALL CSMALL(CLOW,CUP,NINTC,C1,DELFLL,IY)
                IF (IY.GT.0) THEN
                      IF(DELFLL.GE.0.0D0) THEN
                         C=C1
                         N1NEG=N1NEG+1
                         GO TO 10
                      ELSE
                         CSTART=YK(N)+50.0D0
                         CDEL=50.0D0
                         CALL CLARGE(CSTART,CDEL,C2)
                         C=C2
                         N2=N2+1
                         I1NMIN=I1NMIN+1
                         GO TO 10
                      END IF
                ELSE
                      CSTART=YK(N)+50.0D0
                      CDEL=50.0D0
                      CALL CLARGE(CSTART,CDEL,C2)
                      C=C2
                      N2=N2+1
                      GO TO 10
                END IF
            END IF
C     ==========================================
      ELSE
C     ==========================================
C-----------------------------------------SEARCH FOR C1 (BELOW YMIN)
      CLOW=YK(1)-0.01D0
      CUP=YK(1)-50.0D0
      NINTC=50
      CALL CSMALL(CLOW,CUP,NINTC,C1,DELFLL,IY)
          IF (IY.GT.0) THEN
                IF(DELFLL.GE.0.0D0) THEN
                   C=C1
                   N1NEG=N1NEG+1
                   GO TO 10
                ELSE
                   CSTART=YK(1)-50.0D0
                   CDEL=-50.0D0
                   CALL CLARGE(CSTART,CDEL,C2)
                   C=C2
```

```
                    N2NEG=N2NEG+1
                    I1NMIN=I1NMIN+1
                    GO TO 10
                  END IF
            ELSE
                  CLOW=YK(N)+0.01D0
                  CUP=YK(N)+50.0D0
                  NINTC=50
                  CALL CSMALL(CLOW,CUP,NINTC,C1,DELFLL,IY)
                  IF (IY.GT.0) THEN
                        IF(DELFLL.GE.0.0D0) THEN
                           C=C1
                           N1=N1+1
                           GO TO 10
                        ELSE
                           CSTART=YK(1)-50.0D0
                           CDEL=-50.0D0
                           CALL CLARGE(CSTART,CDEL,C2)
                           C=C2
                           N2NEG=N2NEG+1
                           I1MIN=I1MIN+1
                           GO TO 10
                        END IF
                  ELSE
                           CSTART=YK(1)-50.0D0
                           CDEL=-50.0D0
                           CALL CLARGE(CSTART,CDEL,C2)
                           C=C2
                           N2NEG=N2NEG+1
                           GO TO 10
                  END IF
               END IF
         END IF
C
 10    DC=C
       CALL RVALUE (DC,DA,DB,DR)
       A=DA
       B=DB
       RETURN
       END
C
C-----------------------------------------------------------------------
C     CSMALL  :   SUBROUTINE TO SEARCH AND FIND THE ROOT 'C1'(IF EXISTS)
C                 IN THE LOWER RANGE
C-----------------------------------------------------------------------
       SUBROUTINE CSMALL(CLOW,CUP,NINTC,C1,DELH,IY)
C
       DOUBLE PRECISION C,CNXT,A,B,R,RNXT,PROD,DDELC,CLOW,CUP,DELH,H1
       DDELC=(CUP-CLOW)/DFLOAT(NINTC)
       IY=0
       C1=0.0
       C=CLOW
       CALL RVALUE(C,A,B,R)
C      CALL ENTR(A,B,C,H1)
C      WRITE(6,*)'|CSMALL| C  = ',C,'   R = ',R,' ENTROPY = ',H1
       CNXT=C
C
       DO 10  I=1,NINTC
       CNXT=CNXT+DDELC
       CALL RVALUE(CNXT,A,B,RNXT)
```

```
C      CALL ENTR(A,B,CNXT,H1)
C      WRITE(6,*)'|CSMALL| C  = ',CNXT,'  R = ',RNXT,' ENTROPY = ',H1
       PROD=R*RNXT
       IF(PROD.LE.0.0D0)GO TO 20
       R=RNXT
       C=CNXT
 10    CONTINUE
       GO TO 30
C
 20    CALL BISECT(C,R,CNXT,RNXT,C1,DELH)
C
       IY=1
 30    RETURN
       END
C
C-----------------------------------------------------------------------
C     BISECT : SUBROUTINE TO FIND THE ROOT 'XROOT' IN THE INTERVAL(X1,X2)
C              IN WHICH F(X) CHANGES SIGN.
C-----------------------------------------------------------------------
       SUBROUTINE BISECT(X1,F1,X2,F2,XROOT,DELH)
C
       DOUBLE PRECISION X1,F1,X2,F2,XNUM,XDENOM,X3,F3,A,B,PROD
       DOUBLE PRECISION H1,H3,DELH
C
       CALL RVALUE(X1,A,B,F1)
C      CALL ENTR(A,B,X1,H1)
C
 20    XNUM=F2*X1-F1*X2
       XDENOM=F2-F1
       X3=XNUM/XDENOM
       CALL RVALUE(X3,A,B,F3)
C
       IF(DABS(F3).LE.1.0D-06)GO TO 10
       PROD=F1*F3
           IF(PROD.GE.0.0D0)THEN
           X1=X3
           F1=F3
           GO TO 20
           ELSE
           X2=X3
           F2=F3
           GOTO 20
           END IF
 10    XROOT=X3
C
C      CALL ENTR(A,B,X3,H3)
C      WRITE(6,*)'              '
C      WRITE(6,*)'|BISECT| C1 = ',X3,'  R = ',F3,' ENTROPY = ',H3
C      WRITE(6,*)'|BISECT| C1 = ',X3,'  A, B = ',A,B
C      WRITE(6,*)'              '
C      DELH=H3-H1
C
       RETURN
       END
C
C-----------------------------------------------------------------------
C    CLARGE : SUBROUTINE TO FIND THE ASYMPTOTIC ROOT 'C2' IN LARGE RANGE
C-----------------------------------------------------------------------
       SUBROUTINE CLARGE(CSTART,CDEL,C2)
C
```

```
      DOUBLE PRECISION C,CDEL,A,B,R,CSTART,H
      C=CSTART
C
 20   CALL RVALUE(C,A,B,R)
C
C     CALL ENTR(A,B,C,H)
C     WRITE(6,*)' |CLARGE|     C  = ',C,'  R = ',R,'  ENTROPY = ',H
C
      IF(DABS(R).LE.1.0D-8)GO TO 10
      C=C+CDEL
      GO TO 20
C
 10   C2=C
C
C     WRITE(6,*)'       '
C     WRITE(6,*)'|CLARGE| C  = ',C,'  R = ',R,'  ENTROPY = ',H
C     WRITE(6,*)'|CLARGE| C  = ',C,'  A, B = ',A,B
C     WRITE(6,*)'       '
C
C     AA=A
C     BB=B
C     CALL ESTLP3(AA,BB,C2,XKM,XKV)
C     WRITE(6,*)'  MLE EST.     ',XKM,XKV
C
      RETURN
      END
C
C-----------------------------------------------------------------------
C     RVALUE : SUBROUTINE TO COMPUTE PARAMETERS A, B AND RESIDUAL R IN
C              MLE EQUATIONS FOR A SPECIFIED LOCATION PARAMETER 'DC'
C-----------------------------------------------------------------------
      SUBROUTINE RVALUE(DC,DA,DB,DR)
C
      DOUBLE PRECISION MMPSI,DA,DB,DC,PSIB,DR,DEL,YKM,VYK
      DOUBLE PRECISION YK,DFN
      DOUBLE PRECISION DS,DT
      COMMON/P3YK/YK(100),YKM,VYK
      COMMON/SIZE/N,FN,DFN
C
      DEL = YKM-DC
      DB = DEL*DEL/VYK
      DA = DEL/DB
      PSIB = MMPSI(DB,IER)
C
      DS = 0.D0
      DO 10 I=1,N
      DT = (YK(I)-DC)/DA
 10   DS = DS + DLOG(DT)
      DR = -DFN*PSIB + DS
      RETURN
      END
C
C-----------------------------------------------------------------------
C    ENTR: SUBROUTINE TO COMPUTE THE ENTROPY FUNCTION H = H(A,B,C)
C-----------------------------------------------------------------------
      SUBROUTINE ENTR(DA,DB,DC,H)
C
      DOUBLE PRECISION DA,DB,DC,H,DLGAMA,DLGAM
      DOUBLE PRECISION PSIB,MMPSI
      DLGAM=DLGAMA(DB)
```

```
      PSIB=MMPSI(DB,IER)
      H = DLOG(DABS(DA)) + DLGAM - (DB-1.D0)*PSIB + DB
      RETURN
      END
C
C
C*************************************************************************
C--LPQNTL - SUB-ROUTINE TO COMPUTE LOG PEARSON QUANTILES. PEARSON
C  FACTORS GIVEN IN WRC BULLETIN #17 (K-TABLES) ARE LINEARLY
C  INTERPOLATED
C*************************************************************************
      SUBROUTINE LPQNTL(XM,STD,SK,XMR)
C
      COMMON/QUANT/Q10,Q25,Q50,Q100,Q200,Q500
      DIMENSION XK(15,111),CDF(15),RTPK(15),RTLF(15),K(15),Q(15),
     1XJ(111),X1(111),X2(111),X3(111),X4(111),X5(111),X6(111),
     1X7(111),X8(111),X9(111),X10(111),X11(111),X12(111),
     1X13(111),X14(111),X15(111)
      REAL K
      DATA CDF/.005,.01,.02,.04,.1,.2,.5,.8,.9,.96,.98,.99,.995,.998,
     1.999/
      DATA X1/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.606,-.625,-.6452,
     3-.6667,-.6896,-.7143,-.7407,-.7691,-.7997,-.8328,-.8686,-.9074,
     4-.9495,-.995,-1.0443,-1.0975,-1.1548,-1.2162,-1.2817,-1.3511,
     5-1.4244,-1.5011,-1.5811,-1.6639,-1.7492,-1.8366,-1.9258,-2.0164,
     6-2.10825,-2.2009,-2.2942,-2.388,-2.4819,-2.5758,
     70.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,
     80.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,25*0./
      DATA X2/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.6061,-.625,-.6451,
     1-0.6666,-0.6896,-.7145,-0.7405,-0.7688,-0.7992,-0.832,-0.8672,
     1-0.9052,-0.9461,-0.99,-1.037,-1.0871,-1.1404,-1.1968,-1.2561,
     2-1.3182,-1.3827,-1.4494,-1.5181,-1.5884,-1.66,-1.7327,-1.8062,
     3-1.8803,-1.9547,-2.0293,-2.1039,-2.1784,-2.2526,-2.3264,55*0./
      DATA X3/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.6061,-.625,-.6451,
     1-0.6665,-0.6894,-.7138,-0.7399,-0.7678,-0.79765,-0.8296,-0.8637,
     1-0.9001,-0.9388,-0.9798,-1.0231,-1.0686,-1.1163,-1.1658,-1.2172,
     1-1.27,-1.3241,-1.3793,-1.4353,-1.4919,-1.5489,-1.606,-1.6633,
     2-1.7203,-1.7772,-1.8336,-1.8896,-1.945,-1.9997,-2.0538,55*0./
      DATA X5/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44443,-.45452,-.4651,-.4761,-.4877,-.4999,
     2-.5126,-.526,-.5401,-.5548,-.57035,-.5867,-.6038,-.62175,-.6406,
     1-0.6602,-0.6808,-.7021,-0.7242,-0.7471,-0.7706,-0.7947,
     1-0.8193,-0.8442,-0.8694,-0.8946,-0.9199,-0.945,-0.9698,-0.9942,
     2-1.0181,-1.0414,-1.0641,-1.0861,-1.1073,-1.1276,-1.1471,-1.1657,
     3-1.1835,-1.2003,-1.2162,-1.2311,-1.2452,-1.2582,-1.2704,-1.2816,
     455*0./
      DATA X6/-.3636,-.3704,-.37734,-.38458,-.39211,-.39993,-.40806,
     1-.4165,-.4253,-.4345,-.444,-.454,-.4643,-.475,-.4862,-.4978,
     2-.5099,-.5224,-.5353,-.5487,-.5624,-.5765,-.591,-.6057,-.6206,
     1-0.6357,-0.6509,-.666,-0.6811,-0.696,-0.7107,-0.725,-0.7388,
     1-0.7521,-0.7648,-0.7769,-0.7882,-0.7987,-0.8084,-0.8172,-0.8252,
     2-.8322,-.8384,-.8437,-.8481,-.8516,-.8543,-.8561,-.857,-.8572,
     3-.8565,-.8551,-.8529,-.8499,-.8461,-.8416,55*0./
      DATA X7/-.3546,-.3596,-.3645,-.3695,-.3743,-.379,-.3836,-.388,
```

```
1-.3922,-.3962,-.3999,-.4032,-.4062,-.4088,-.411,-.4127,-.4138,
2-.4144,-.4144,-.4138,-.4125,-.4106,-.4079,-.4045,-.4004,
1-.3955,-.3899,-.3835,-.3764,-.3685,-.3599,-.3506,-.3406,
1-.33,-.3187,-.3069,-.2944,-.2815,-.2681,-.2542,-.24,-.2254,
2-.2104,-.1952,-.1797,-.164,-.1481,-.132,-.1158,-.0995,-.083,
3-.0665,-.0499,-.0333,-.0166,0.,55*0./
 DATA X8/-.0103,.00243,.0156,.0293,.0434,.058,.073,.0885,.1044,
1.1207,.1374,.1545,.1719,.1897,.2078,.2262,.2448,.2638,.2829,.3022,
2.3217,.3413,.361,.3808,.4006,
1.4204,.4402,.4598,.4793,.4987,.5179,.5368,.5555,.5738,
1.5918,.6094,.6266,.6434,.6596,.6753,.6905,.7051,.7192,.7326,
2.7454,.7575,.769,.7799,.79,.7995,.8083,.8164,.8238,.8304,.8364,
3.8416,55*0./
 DATA X9/.6912,.712,.7328,.7536,.7746,.7955,.8164,.8373,.8582,
1.879,.8996,.9202,.9406,.9609,.981,1.0008,1.0204,1.0397,1.0586,
21.0773,1.0955,1.1134,1.1308,1.1477,1.1642,1.1801,1.1954,
11.2101,1.2242,1.2377,1.2504,1.2624,1.2737,1.2841,1.2938,
11.3026,1.3105,1.3176,1.3238,1.329,1.3333,1.3367,1.339,1.3405,
21.3409,1.3404,1.3389,1.3364,1.3329,1.3285,1.3231,1.3167,1.3094,
31.3011,1.2918,1.2816,55*0./
 DATA X10/2.0474,2.0637,2.0795,2.0949,2.1099,2.1243,2.1383,2.1517,
12.1647,2.177,2.1887,2.1999,2.2104,2.2202,2.2294,2.2379,2.2456,
22.2525,2.2587,2.2641,2.2686,2.2723,2.2751,2.2769,2.2779,2.2778,
12.27676,2.2747,2.2716,2.2674,2.2622,2.2558,2.2483,2.2397,
12.2299,2.2189,2.2067,2.1933,2.1787,2.1629,2.1459,2.1277,2.1082,
22.0876,2.0657,2.0427,2.0185,1.9931,1.9666,1.939,1.9102,1.8804,
31.8495,1.8176,1.7846,1.7507,1.7158,1.68,1.6433,1.6057,1.5674,
41.5283,1.4885,1.4481,1.4072,1.3658,1.3241,1.2823,1.2403,1.1984,
51.1568,1.1157,1.0751,1.0354,.9967,.9592,.923,.8881,.8549,.8232,
6.7931,.7646,.7377,.7123,.6884,.6659,
6.6447,.6247,.6059,.5881,.5714,.5555,.5405,.5263,.5128,.5,.4878,
7.4762,.4651,.4546,.4444,.4348,.4255,.4167,.4082,.4,.3922,.3846,
8.3774,.3704,.3636/
 DATA X11/3.2838,3.2884,3.2924,3.2957,3.2982,3.30007,3.3012,
13.3015,3.301,3.2998,3.2977,3.2947,3.2909,3.2862,3.2806,3.274,
23.2665,3.258,3.2485,3.238,3.2264,3.2138,3.2,3.1851,3.1691,
33.1519,3.1336,3.114,3.0932,3.0712,3.0479,3.0233,2.9974,2.9703,
42.9418,2.912,2.8809,2.8485,2.8147,2.7796,2.7433,2.7056,2.6666,
52.6263,2.5848,2.5421,2.4981,2.453,2.4067,2.3593,2.3108,2.2613,
62.2108,2.1594,2.107,2.0538,55*0./
 DATA X12/4.6402,4.6285,4.6159,4.6025,4.5882,4.573,4.5569,4.5399,
14.5219,4.503,4.483,4.4621,4.4401,4.4171,4.393,4.3678,4.3415,4.314,
24.2855,4.2557,4.2247,4.1926,4.1592,4.1245,4.0886,
34.0514,4.0129,3.973,3.9318,3.8893,3.8454,3.8001,3.7535,3.7054,
43.656,3.6052,3.553,3.4994,3.4444,3.388,3.3304,3.2713,3.211,
53.1494,3.0866,3.0226,2.9574,2.891,2.8236,2.7551,2.6857,2.6154,
62.5442,2.4723,2.3996,2.3264,55*0./
 DATA X13/6.08307,6.0517,6.0193,5.986,5.9517,5.9164,5.88,5.8427,
15.8042,5.7646,5.724,5.6822,5.6393,5.5953,5.5501,5.5036,5.456,
25.4071,5.357,5.3056,5.2529,5.1989,5.1436,5.087,5.029,
34.9696,4.9088,4.8467,4.7831,4.7182,4.6518,4.5839,4.5147,4.444,
44.3719,4.2983,4.2234,4.147,4.0693,3.9902,3.9097,3.828,3.745,
53.6607,3.5753,3.4887,3.4011,3.3124,3.2228,3.1323,3.041,2.949,
62.8564,2.7632,2.6697,2.5758,55*0./
 DATA X14/8.0869,8.0259,7.9639,7.9008,7.8366,7.7712,7.7048,
17.6372,7.5684,7.4985,7.4273,7.355,7.2814,7.2065,7.1304,7.053,
26.9744,6.8944,6.813,6.7303,6.6463,6.5608,6.474,6.3858,6.2961,
16.20506,6.1125,6.0186,5.9232,5.8263,5.728,5.6282,5.5269,
15.4243,5.3201,5.2146,5.1077,4.9994,4.8897,4.7788,4.6665,4.553,
24.4384,4.3226,4.2058,4.088,3.9693,3.8498,3.7296,3.6087,3.4874,
```

```
     33.3657,3.2437,3.1217,2.9998,2.8782,2.7571,2.6367,2.5174,
     42.3994,2.2831,2.1688,2.057,1.9481,1.8424,1.7406,1.6431,1.5502,
     51.4623,1.3798,1.3028,1.2313,1.1653,1.1047,1.049,.998,.9513,.9085,
     6.8693,.8332,.7999,.7692,.7407,.7143,.6896,.6667,25*0./
      DATA X15/9.6577,9.5723,9.4859,9.3983,9.3095,9.2196,9.1285,9.0362,
     18.9427,8.848,8.752,8.6548,8.5563,8.4565,8.3553,8.2529,8.1491,
     28.044,7.9374,7.8295,7.7202,7.6095,7.4974,7.3838,7.2688,
     17.1524,7.0344,6.9151,6.7942,6.6719,6.5481,6.4229,6.2963,
     26.1682,6.0387,5.9078,5.7755,5.6419,5.507,5.3709,5.2335,5.0951,
     34.9555,4.8149,4.6734,4.5311,4.3881,4.2444,4.1002,3.9557,3.8109,
     43.6661,3.5214,3.377,3.2332,3.0902,2.9483,2.8079,2.6692,2.5326,
     52.3987,2.2678,2.1405,2.0174,1.8989,1.7857,1.6783,1.577,1.4822,
     61.3941,1.3128,1.2381,1.1697,1.1074,1.0507,.999,.9519,.9089,.8695,
     7.8333,.8,.7692,.7407,.7143,.6897,.6667,25*0./
      J=111
      DO 61 I=1,25
      X14(J)=-X1(I)
      X15(J)=-X1(I)
61    J=J-1
      J=111
      DO 62 I=1,111
      X4(J)=-X10(I)
62    J=J-1
      J=111
      DO 63 I=1,55
      X1(J)=-X13(I)
      X2(J)=-X12(I)
      X3(J)=-X11(I)
      X5(J)=-X9(I)
      X6(J)=-X8(I)
      X7(J)=-X7(I)
      X8(J)=-X6(I)
      X9(J)=-X5(I)
      X11(J)=-X3(I)
      X12(J)=-X2(I)
      X13(J)=-X1(I)
63    J=J-1
      DO 1 J =1,111
      XK(1,J)=X1(J)
      XK(2,J)=X2(J)
      XK(3,J)=X3(J)
      XK(4,J)=X4(J)
      XK(5,J)=X5(J)
      XK(6,J)=X6(J)
      XK(7,J)=X7(J)
      XK(8,J)=X8(J)
      XK(9,J)=X9(J)
      XK(10,J)=X10(J)
      XK(11,J)=X11(J)
      XK(12,J)=X12(J)
      XK(13,J)=X13(J)
      XK(14,J)=X14(J)
      XK(15,J)=X15(J)
 1    CONTINUE
      DO 65 I=1,15
      RTLF(I)=1./CDF(I)
65    RTPK(I)=1./(1.-CDF(I))
      RTPK(15)=1000.
      RTPK(14)=500.
      RTPK(13)=200.
```

```
      RTPK(12)=100.
      J=1
 301  W=J
      XJ(J)=5.6-W/10.0
      IF(XJ(J)-SK)303,303,302
 302  J=J+1
      GO TO 301
 303  DO 304 I=9,14
      VK =((SK-XJ(J))*(XK(I,J-1)-XK(I,J)))/(XJ(J-1)-XJ(J))+XK(I,J)
      K(I)=EXP(XM+VK*STD)
 304  CONTINUE
      DO 305 I=9,14
 305  Q(I)=K(I)*XMR
      Q10=Q(9)
      Q25=Q(10)
      Q50=Q(11)
      Q100=Q(12)
      Q200=Q(13)
      Q500=Q(14)
C
C     WRITE(6,310)
C310  FORMAT(/' *** LOG PEARSON VARIATE ESTIMATES BY MXM1 METHOD ***'//
C    19X,'CDF',2X,'T(FOR LOS)',2X,'T(FOR PKS)',5X,'VARIATE',//)
C     DO 315 I=1,15
C     WRITE(6,320) CDF(I),RTLF(I),RTPK(I),Q(I)
C315  CONTINUE
C     WRITE(18,330)Q(7),Q(8),Q(9),Q(11),Q(12),Q(13),Q(14)
C330  FORMAT(3X,'MOMIX',1X,7(2X,F7.1))
C320  FORMAT(3F12.3,F12.2)
C     WRITE(6,325)
C325  FORMAT(' NOTE: T=RETURN PERIOD(YRS), LOS=MINIMUM VALUES LIKE LOW F
C    1LOWS, PKS=MAXIMUM VALUES LIKE FLOOD FLOWS')
      RETURN
      END
C
C
C-----------------------------------------------------------------------
C     SEARCH : SUBROUTINE TO SEARCH FOR THE ROOT OF 'C' IN BOTH +VE
C              AND -VE RANGE OF SMALL AS WELL AS LARGE VALUES.
C
      SUBROUTINE SEARCH(CP,C1,C2,C2NEG,FLLD1,FLLD2,FLLD2N)
      DOUBLE PRECISION YK,CLOW,CUP,CSTART,CDEL,DFN,YKM,VYK
      DOUBLE PRECISION FLLD1,FLLD2,FLLD2N
      COMMON/SIZE/N,FN,DFN
      COMMON/P3YK/YK(100),YKM,VYK
      C1=0.0
      C2=0.0
      C2NEG=0.0
      FLLD1=-1000.0D0
      FLLD2=-1000.0D0
      FLLD2N=-1000.0D0
      IF(CP.GT.0.0)THEN
C-----------------------------------------SEARCH FOR C1 (ABOVE YMAX)
      CLOW=YK(N)+0.01D0
      CUP=YK(N)+50.0D0
      NINTC=50
      CALL CSMALL(CLOW,CUP,NINTC,C1,FLLD1)
           IF (C1.NE.0.0) THEN
C-------------------------------------------------------FIND C2NEG
           CSTART=YK(1)-50.0D0
```

```
            CDEL=-50.0D0
            CALL CLARGE(CSTART,CDEL,C2NEG,FLLD2N)
            RETURN
            ELSE
C-----------------------------------------------------------------FIND C2
            CSTART=YK(N)+50.0D0
            CDEL=50.0D0
            CALL CLARGE(CSTART,CDEL,C2,FLLD2)
C-----------------------------------------SEARCH FOR C1 (BELOW YMIN)
            CLOW=YK(1)-0.01D0
            CUP=YK(1)-50.0D0
            NINTC=50
            CALL CSMALL(CLOW,CUP,NINTC,C1,FLLD1)
                 IF (C1.NE.0.0) THEN
                 RETURN
                 ELSE
C-----------------------------------------------------------FIND C2NEG
                 CSTART=YK(1)-50.0D0
                 CDEL=-50.0D0
                 CALL CLARGE(CSTART,CDEL,C2NEG,FLLD2N)
                 RETURN
                 END IF
            END IF
C     =====
      ELSE
C     =====
C-----------------------------------------SEARCH FOR C1 (BELOW YMIN)
      CLOW=YK(1)-0.01D0
      CUP=YK(1)-50.0D0
      NINTC=50
      CALL CSMALL(CLOW,CUP,NINTC,C1,FLLD1)
            IF (C1.NE.0.0) THEN
C-----------------------------------------------------------------FIND C2
            CSTART=YK(N)+50.0D0
            CDEL=50.0D0
            CALL CLARGE(CSTART,CDEL,C2,FLLD2)
            RETURN
            ELSE
C-----------------------------------------------------------FIND C2NEG
            CSTART=YK(1)-50.0D0
            CDEL=-50.0D0
            CALL CLARGE(CSTART,CDEL,C2NEG,FLLD2N)
C-----------------------------------------SEARCH FOR C1 (ABOVE YMAX)
            CLOW=YK(N)+0.01D0
            CUP=YK(N)+50.0D0
            NINTC=50
            CALL CSMALL(CLOW,CUP,NINTC,C1,FLLD1)
                 IF (C1.NE.0.0) THEN
                 RETURN
                 ELSE
C-----------------------------------------------------------------FIND C2
                 CSTART=YK(N)+50.0D0
                 CDEL=50.0D0
                 CALL CLARGE(CSTART,CDEL,C2,FLLD2)
                 RETURN
                 END IF
            END IF
      END IF
      RETURN
      END
```

```
C
C-----------------------------------------------------------------
C   PARMS : PICKS THE LOCATION PARM. 'C' CORRESPONDING TO MAXIMUM VALUE
C           OF THE LOG-LIKELIHOOD FUNCTION. COMPUTES PARMS. A & B
C
      SUBROUTINE PARMS(C1,C2,C2NEG,FLLD1,FLLD2,FLLD2N,C,A,B)
      DOUBLE PRECISION DC,DA,DB,DR
      DOUBLE PRECISION FLLD1,FLLD2,FLLD2N
      IF(FLLD1.GE.FLLD2) GO TO 10
           IF(FLLD2.GE.FLLD2N) GO TO 20
                C=C2NEG
                DC=C
                CALL RVALUE(DC,DA,DB,DR)
                A=DA
                B=DB
                RETURN
C
  20            C=C2
                DC=C
                CALL RVALUE(DC,DA,DB,DR)
                A=DA
                B=DB
                RETURN
C
  10  IF(FLLD1.GE.FLLD2N) GO TO 30
                C=C2NEG
                DC=C
                CALL RVALUE(DC,DA,DB,DR)
                A=DA
                B=DB
                RETURN
C
  30            C=C1
                DC=C
                CALL RVALUE(DC,DA,DB,DR)
                A=DA
                B=DB
                RETURN
      END
C
C*********************************************************************
C    SERCH2 : SUBROUTINE TO SEARCH FOR TWO ROOTS OF THE LOCATION PARM.
C          'C' IN LOWER AND HIGHER REGION (REF : RAO'S MLE PAPER, 1986)
C*********************************************************************
      SUBROUTINE SERCH2(CSTART,CDEL,C1,C2)
      DOUBLE PRECISION DA,DB,DC,DR,CSTART,CDEL
      DOUBLE PRECISION FLIKE
      DC=CSTART
      ITR=1
C
 13   CALL RVALUE(DC,DA,DB,DR)
C     CALL FNLKD(DA,DB,DC,FLIKE)
C
      WRITE(6,*)DC,DR,FLIKE
      DC=DC+CDEL
      IF(ITR.GE.25)GO TO 12
      ITR=ITR+1
      GO TO 13
C
 12   RETURN
```

```
      END
C
C
C*************************************************************************
C     QNTL : SUBROUTINE TO FIND THE QUANTILE BY CONVERTING THE LP3
C            VARIATE TO THE STANDARDIZED GAMMA VARIATE
C*************************************************************************
C
      SUBROUTINE QNTL(A,B,C,XM)
      DIMENSION RTP(6),EXPROB(6),Q(6)
      COMMON/QUANT/Q10,Q25,Q50,Q100,Q200,Q500
      DOUBLE PRECISION FNGAMA,B1
      RTP(1)=10.
      RTP(2)=25.
      RTP(3)=50.
      RTP(4)=100.
      RTP(5)=200.
      RTP(6)=500.
      DO 5 I=1,6
      EXPROB(I)=1./RTP(I)
    5 CONTINUE
      IF(A.GT.0.0) THEN
          DO 10 I=1,6
          XT=XM
          T=(ALOG(XT)-C)/A
   30     CALL MDGAM(T,B,CUMF,IER)
          G=EXPROB(I)-(1.-CUMF)
          IF(G.LE.1.0E-06)GO TO 20
          FDENST=EXP(-T)*(T**(B-1.))/FNGAMA(B)
          T=T-G/FDENST
          GO TO 30
   20     XT=EXP(A*T+C)
          Q(I)=XT*XM
   10     CONTINUE
      ELSE
          DO 40 I=1,6
          XT=XM
          T=(ALOG(XT)-C)/A
   60     CALL MDGAM(T,B,CUMF,IER)
          G=CUMF-EXPROB(I)
          IF(G.LE.1.0E-06)GO TO 50
          B1=B
          FDENST=EXP(-T)*(T**(B-1.))/FNGAMA(B1)
          T=T-G/FDENST
          GO TO 60
   50     XT=EXP(A*T+C)
          Q(I)=XT*XM
   40     CONTINUE
      END IF
      RETURN
      END
C
C     FNGAMA : USED BY THE SUBROUTINE 'QNTL' ABOVE
C
      FUNCTION FNGAMA(B)
      DOUBLE PRECISION B,B1,PROD,FNGAMA
      B1=B
      PROD=1.0D0
   10 B1=B1-1.D0
      PROD=PROD*B1
```

```
      IF(B1.LT.57.)THEN
          PROD=PROD*DGAMMA(B1)
          FNGAMA=PROD
          RETURN
      ELSE
          B1=B1-1.D0
          GO TO 10
      END IF
      RETURN
      END
C
C*************************************************************************
C     UBVSK: SUB-ROUTINE TO COMPUTE UNBIASED MEAN, VARIANCE, &
C     SKEWNESS COEFFICIENT (VM,VV,VSK RESPECTIVELY)
C*************************************************************************
C
      SUBROUTINE UBVSK(V,VM,VV,VSK)
      DOUBLE PRECISION DFN
      COMMON/SIZE/N,FN,DFN
      DIMENSION V(N)
C
      C1=FN/(FN-1.)
      C2=FN**2/(FN-1.)/(FN-2.)
      C2=C2/C1**1.5
C
      X1=0.
      X2=0.
      X3=0.
      DO 10 I=1,N
      V1=V(I)
      V2=V1*V1
      V3=V2*V1
C
      X1=X1+V1
      X2=X2+V2
 10   X3=X3+V3
C
      VM=X1/FN
      VV=X2/FN-VM**2
      VSK=(X3/FN-3.*VM*VV-VM**3)/VV**1.5
      VV=VV*C1
      VSK=VSK*C2
      RETURN
      END
C
C
      SUBROUTINE ESTLP3(A,B,C,XM,XV)
      DOUBLE PRECISION DA,DB,DC,DA1,DA2,DAL1,DAL2
      DOUBLE PRECISION XMLN,DXM,CV2LN1,CV2,DXV
      WRITE(6,*)' |ESTLP3|  A, B, C = ',A,B,C
      XM=0.0
      XV=0.0
      DA=A
      DB=B
      DC=C
      IF(DA.GE.1.D0) GO TO 10
      DA1=1.D0-DA
      DAL1=DLOG(DA1)
C
      XMLN=DC-DB*DAL1
```

```
      DXM=DEXP(XMLN)
      XM=DXM
C
      IF(DA.GE.0.5D0)GO TO 10
      DA2=1.D0-(2.D0*DA)
      DAL2=DLOG(DA2)
      CV2LN1=DB*(2.0D0*DAL1-DAL2)
      CV2=DEXP(CV2LN1)-1.0D0
      DXV=CV2*DXM*DXM
      XV=DXV
C
   10 RETURN
      END
$ENTRY
//GO.FT10F001 DD DSN=CEAROR.LP3.OUTPUTM8,DISP=SHR
$$
//
//GO.FT10F001 DD DSN=CEAROR.LP3.OUTPUTM7,DISP=SHR
```

```
//ESTIMAT1 JOB (1304,59634,1,20),'ARORA',MSGCLASS=S,CLASS=Q              00010000
/*ROUTE PRINT CEBA                                                      00020000
/*JOBPARM SHIFT=N
//     EXEC WATFIV,REGION.GO=4000K,TIME.GO=99
$JOB            TIME=4500
C
C=====================================================================
C                  LOG PEARSON TYPE 3 DISTRIBUTION
C---------------------------------------------------------------------
C               *** COPIED FROM LP3.SIMU(ESTIMATE) ***
C     THIS PROGRAM CALCULATES THE MEAN AND STD. DEVIATION OF THE MONTE-
C     CARLO SAMPLE ESTIMATES OF THE PARAMETERS, SKEWNESS RATIO, AND
C     QUANTILES.  THESE SAMPLE ESTIMATES, PREVIOUSLY CALCULATED, ARE
C     READ FROM DATA SET : LP3.OUTPUTM? WHERE ? DEPENDS ON THE METHOD.
C=====================================================================
C
      DOUBLE PRECISION DSEED
      DIMENSION A(1000),B(1000),C(1000),SKXP(1000),Q10(1000)
      DIMENSION Q25(1000),Q50(1000),Q100(1000),Q200(1000),Q500(1000)
C
      READ(10,*)AP,BP,CP,COVP,SKP,DSEED
      WRITE(11,*)AP,BP,CP,COVP,SKP
      WRITE(12,*)AP,BP,CP,COVP,SKP
      WRITE(13,*)AP,BP,CP,COVP,SKP
      WRITE(6,5)DSEED
  5   FORMAT(1H1,' DSEED = ',D28.16)
      WRITE(6,*)' POPULATION PARAMETERS A, B, C & C.V.,SKEW :'
      WRITE(6,*)AP,BP,CP,COVP,SKP
C
      DO 10 I=1,5
      READ(10,*)N,M
      WRITE(6,*)'                                                    '
      WRITE(6,*)'************************************************************
     @************************************************'
      WRITE(6,*)'     SAMPLE SIZE = ',N,'   NO. OF SAMPLES = ',M
      WRITE(6,*)'                                                    '
C
      DO 20 J=1,M
      READ(10,1)JJ,A(J),B(J),C(J),SKXP(J),Q10(J),Q25(J),Q50(J),Q100(J),
     *Q200(J),Q500(J)
   1  FORMAT(1X,I4,F16.7,E15.7,F17.7,F13.7,6F10.5)
  20  CONTINUE
C
      CALL ESTST(A,M,AM,AST,AMIN,AMAX)
      CALL ESTST(B,M,BM,BST,BMIN,BMAX)
      CALL ESTST(C,M,CM,CST,CMIN,CMAX)
      WRITE(6,*)'                                                    '
      WRITE(6,*)N,' MIN. A = ',AMIN,'   MAX. A = ',AMAX
      WRITE(6,*)N,' MIN. B = ',BMIN,'   MAX. B = ',BMAX
      WRITE(6,*)N,' MIN. C = ',CMIN,'   MAX. C = ',CMAX
      WRITE(6,*)'                                                    '
      WRITE(6,*)' PARAMETER STATS - MEAN/STD. DEV :'
      WRITE(6,*)'=================================='
      WRITE(6,*)'             A                  B                  C'
```

```
      WRITE(6,4)N,AM,BM,CM
      WRITE(6,4)N,AST,BST,CST
    4 FORMAT(1X,I4,F15.7,F20.7,F17.7)
      WRITE(11,2)N,AM,AST,BM,BST,CM,CST
    2 FORMAT(I4,2F15.7,2F20.7,2F17.7)
      WRITE(6,*)'                                                    '
C
      CALL ESTST(Q10,M,Q10M,Q10ST,Q10MN,Q10MX)
      CALL ESTST(Q25,M,Q25M,Q25ST,Q25MN,Q25MX)
      CALL ESTST(Q50,M,Q50M,Q50ST,Q50MN,Q50MX)
      CALL ESTST(Q100,M,Q100M,Q100ST,Q100MN,Q100MX)
      CALL ESTST(Q200,M,Q200M,Q200ST,Q200MN,Q200MX)
      CALL ESTST(Q500,M,Q500M,Q500ST,Q500MN,Q500MX)
      WRITE(6,*)'                                                    '
      WRITE(6,*)N,'  MIN. Q10 = ',Q10MN,'    MAX. Q10 = ',Q10MX
      WRITE(6,*)N,'  MIN. Q25 = ',Q25MN,'    MAX. Q25 = ',Q25MX
      WRITE(6,*)N,'  MIN. Q50 = ',Q50MN,'    MAX. Q50 = ',Q50MX
      WRITE(6,*)N,' MIN. Q100 = ',Q100MN,'   MAX. Q100 = ',Q100MX
      WRITE(6,*)N,' MIN. Q200 = ',Q200MN,'   MAX. Q200 = ',Q200MX
      WRITE(6,*)N,' MIN. Q500 = ',Q500MN,'   MAX. Q500 = ',Q500MX
C
      WRITE(6,*)'                                                    '
      WRITE(6,*)'QUANTILE STATS - MEAN/STD. DEV. :'
      WRITE(6,*)'═══════════════════════════════'
      WRITE(6,*)'            Q10            Q25            Q50
     *  Q100           Q200           Q500'
      WRITE(6,*)'                                                    '
      WRITE(6,3)N,Q10M,Q25M,Q50M,Q100M,Q200M,Q500M
      WRITE(6,3)N,Q10ST,Q25ST,Q50ST,Q100ST,Q200ST,Q500ST
      WRITE(12,3)N,Q10M,Q25M,Q50M,Q100M,Q200M,Q500M
      WRITE(13,3)N,Q10ST,Q25ST,Q50ST,Q100ST,Q200ST,Q500ST
    3 FORMAT(I4,6F15.7)
      WRITE(6,*)'                                                    '
C
   10 CONTINUE
      STOP
      END
C
C
      SUBROUTINE ESTST(X,M,XM,XSTD,XMIN,XMAX)
      DIMENSION X(M)
      A=0.
      B=0.
      XMIN=X(1)
      XMAX=X(2)
      DO 10 I=1,M
      XMIN=AMIN1(X(I),XMIN)
      XMAX=AMAX1(X(I),XMAX)
      A=A+X(I)
      B=B+X(I)*X(I)
   10 CONTINUE
      C1=FLOAT(M)
      C2=C1/(C1-1.)
      XM=A/C1
      XVAR=(B/C1-XM*XM)*C2
      XSTD=SQRT(XVAR)
C     WRITE(6,*)M,' MEAN = ',XM,'|   XSTD = ',XSTD
      RETURN
      END
C
```

```
      SUBROUTINE SKEWX(A,B,C,SKX)
      DOUBLE PRECISION DA,DB,DC,DVARX,DSKX
      DOUBLE PRECISION DA1,DA2,DA3
      DOUBLE PRECISION DA1B,DA1BI,DA12B,DA12BI,DA13B,DA13BI
      DOUBLE PRECISION DA2B,DA2BI,DA3B,DA3BI
      DA=A
      DB=B
      DC=C
C
      DA1=1.D0-DA
      DA2=DA1-DA
      DA3=DA2-DA
C
      DA1B=DA1**DB
      DA1BI=1.D0/DA1B
      DA12B=DA1B**2.D0
      DA12BI=1.D0/DA12B
      DA13B=DA1B**3.D0
      DA13BI=1.D0/DA13B
C
      DA2B=DA2**DB
      DA2BI=1.D0/DA2B
C
      DA3B=DA3**DB
      DA3BI=1.D0/DA3B
C
      DVARX=DEXP(2.D0*DC)*(DA2BI-DA12BI)
      DSKX=DEXP(3.D0*DC)*(DA3BI-3.D0*DA2BI*DA1BI+2.D0*DA13BI)
      DSKX=DSKX/(DVARX**0.15D01)
      SKX=DSKX
      RETURN
      END
$ENTRY
//GO.FT10F001 DD DSN=CEAROR.LP3.OUTPUTM8,DISP=SHR
//GO.FT11F001 DD DSN=CEAROR.LP3.M7PARM,DISP=SHR
//GO.FT12F001 DD DSN=CEAROR.LP3.M7BSQN,DISP=SHR
//GO.FT13F001 DD DSN=CEAROR.LP3.M7STQN,DISP=SHR
$$
//
```

```
//AAAAAAA JOB (1304,59634,5,20),'ARORA',MSGCLASS=S,CLASS=Q                00010000
/*ROUTE PRINT CEBA                                                        00020000
/*JOBPARM SHIFT=N
//     EXEC WATFIV,LIB='CEAROR.SPEC.LIB',REGION.GO=4000K,TIME.GO=99
$JOB            TIME=4500,NOEXT
C
C=================================================================
C                  LOG PEARSON TYPE 3 DISTRIBUTION
C-----------------------------------------------------------------
C     PROGRAM TO COMPUTE THE RELATIVE BIAS (BIAS), STANDARD ERROR (SE),
C           AND ROOT MEAN SQUARE ERROR (RMSE) OF QUANTILES
C=================================================================
C
      REAL MSQ10,MSQ25,MSQ50,MSQ100,MSQ200,MSQ500
      CHARACTER *1 DUMMY
      CHARACTER *4 NAME(6)
      NCASE = 5
      CV    = 0.7
      SK    = 3.
      NAME(1) = 'MMD '
      NAME(2) = 'MMI1'
      NAME(3) = 'MMI2'
      NAME(4) = 'MIX '
      NAME(5) = 'MLE '
      NAME(6) = 'ENT '
C
C
      WRITE(11,1)
      WRITE(12,2)
      WRITE(13,3)
  1   FORMAT(////////,15X,72('='),//,
     * 15X,23X,'BIAS OF SELECTED QUANTILES',/)
  2   FORMAT(////////,15X,72('='),//,
     * 15X,18X,'STANDARD ERROR OF SELECTED QUANTILES',/)
  3   FORMAT(////////,15X,72('='),//,
     * 15X,14X,'ROOT MEAN SQUARE ERROR OF SELECTED QUANTILES',/)
C
      WRITE(11,4)NCASE,CV,SK
      WRITE(12,4)NCASE,CV,SK
      WRITE(13,4)NCASE,CV,SK
  4   FORMAT(15X,15X,'( CASE - ',I2,3X,' C.V. = ',F4.1,3X,'SKEW = ',
     * F4.1,1X,')',/,15X,72('-'),/,
     5 15X,38X,'RETURN PERIOD',/,
     6 15X,7X,'SAMPLE',2X,57('-'),/,
     7 15X,'METHOD',2X,'SIZE','     10        25        50        100
     7    200       500'
     8 ,/,15X,72('='))
C
C
      READ(9,5)Q10P,Q25P,Q50P,Q100P,Q200P,Q500P
  5   FORMAT(////////////,3X,6F15.5,/)
      WRITE(6,*)Q10P,Q25P,Q50P,Q100P,Q200P,Q500P
      READ(10,6)
  6   FORMAT(//////////)
```

```
C
      DO 10 I=1,5
      DO 20 J=1,6
      READ(9,7)ISIZE,Q10M,Q25M,Q50M,Q100M,Q200M,Q500M
      READ(10,7)ISIZE,Q10ST,Q25ST,Q50ST,Q100ST,Q200ST,Q500ST
      WRITE(6,*)ISIZE,Q10M,Q25M,Q50M,Q100M,Q200M,Q500M
      WRITE(6,*)ISIZE,Q10ST,Q25ST,Q50ST,Q100ST,Q200ST,Q500ST
  7   FORMAT(3X,I2,F14.7,5F15.7)
            BQ10 = (Q10M-Q10P)/Q10P
            BQ25 = (Q25M-Q25P)/Q25P
            BQ50 = (Q50M-Q50P)/Q50P
            BQ100 = (Q100M-Q100P)/Q100P
            BQ200 = (Q200M-Q200P)/Q200P
            BQ500 = (Q500M-Q500P)/Q500P
      WRITE(11,8)NAME(J),ISIZE,BQ10,BQ25,BQ50,BQ100,BQ200,BQ500
  8   FORMAT(15X,1X,A4,4X,I2,6(3X,F7.3))
            SEQ10 = Q10ST/Q10P
            SEQ25 = Q25ST/Q25P
            SEQ50 = Q50ST/Q50P
            SEQ100 = Q100ST/Q100P
            SEQ200 = Q200ST/Q200P
            SEQ500 = Q500ST/Q500P
      WRITE(12,8)NAME(J),ISIZE,SEQ10,SEQ25,SEQ50,SEQ100,SEQ200,SEQ500
            MSQ10 = ( 0.999*SEQ10*SEQ10 + BQ10*BQ10 ) ** 0.5
            MSQ25 = ( 0.999*SEQ25*SEQ25 + BQ25*BQ25 ) ** 0.5
            MSQ50 = ( 0.999*SEQ50*SEQ50 + BQ50*BQ50 ) ** 0.5
            MSQ100 = ( 0.999*SEQ100*SEQ100 + BQ100*BQ100 ) ** 0.5
            MSQ200 = ( 0.999*SEQ200*SEQ200 + BQ200*BQ200 ) ** 0.5
            MSQ500 = ( 0.999*SEQ500*SEQ500 + BQ500*BQ500 ) ** 0.5
      WRITE(13,8)NAME(J),ISIZE,MSQ10,MSQ25,MSQ50,MSQ100,MSQ200,MSQ500
 20   CONTINUE
      READ(9,9)DUMMY
      READ(10,9)DUMMY
 9    FORMAT(1X,A1)
      WRITE(11,*)'     '
      WRITE(12,*)'     '
      WRITE(13,*)'     '
 10   CONTINUE
      STOP
      END
$ENTRY
//GO.FT09F001 DD DSN=CEAROR.LP3.BSQN5,DISP=SHR
//GO.FT10F001 DD DSN=CEAROR.LP3.STQN5,DISP=SHR
//GO.FT11F001 DD DSN=CEAROR.LP3.BQ5,DISP=SHR
//GO.FT12F001 DD DSN=CEAROR.LP3.SEQ5,DISP=SHR
//GO.FT13F001 DD DSN=CEAROR.LP3.MSQ5,DISP=SHR
$$
//
//GO.FT10F001 DD DSN=CEAROR.LP3.OUTPUTM7,DISP=SHR
```

```
//PROJECT JOB (1304,59634,1,20),'ARORA',MSGCLASS=S,CLASS=B
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=N
//     EXEC WATFIV,LIB='CEAROR.SPEC.LIB',REGION.GO=4000K,TIME.GO=99
$JOB            TIME=4500,NOEXT
C
C=====================================================================
C                    LOG PEARSON TYPE 3 DISTRIBUTION
C---------------------------------------------------------------------
C     THIS PROGRAM CALCULATES THE LP3 QUANTILES FOR VARIOUS POPULATION
C     PARAMETER SETS
C=====================================================================
C
      COMMON/QUANT/Q10,Q25,Q50,Q100,Q200,Q500
C
      WRITE(6,*)'                '
      WRITE(6,*)'                '
      WRITE(6,*)'          A           B              C           C.V.
     # SKEW      Q10       Q25       Q50       Q100      Q200      Q500'
      WRITE(6,*)'                '
      WRITE(10,*)'          A           B              C           C.V.
     # SKEW      Q10       Q25       Q50       Q100      Q200      Q500'
      WRITE(10,*)'                '
      DO 10 I=1,9
      READ(5,*)COVP,SKP,A,B,C
      XMR=1.0
      CPMY=C+A*B
      VARL=B*A*A
      STDL=SQRT(VARL)
      SKL=2.*(ABS(A)/A)*(1./SQRT(B))
      IF(ABS(SKL).LE.5.5)GO TO 30
      WRITE(6,*)' LOG SKEW = ',SKL
      GO TO 40
  30  CALL LPQNTL(CPMY,STDL,SKL,XMR)
C
  40  WRITE(6,11)A,B,C,COVP,SKP,Q10,Q25,Q50,Q100,Q200,Q500
      WRITE(10,11)A,B,C,COVP,SKP,Q10,Q25,Q50,Q100,Q200,Q500
  11  FORMAT(3F14.6,'  |',2F8.2,'  |',6F10.5)
      X=FLOAT(I)/3.0
      IX=X
      XX=X-FLOAT(IX)
      IF(XX.NE.0.0)GO TO 10
      WRITE(6,*)'      '
      WRITE(10,*)'      '
  10  CONTINUE
      STOP
      END
C
C
C*********************************************************************
C--LPQNTL - SUB-ROUTINE TO COMPUTE LOG PEARSON QUANTILES. PEARSON
C  FACTORS GIVEN IN WRC BULLETIN #17 (K-TABLES) ARE LINEARLY
C  INTERPOLATED
C*********************************************************************
```

```
      SUBROUTINE LPQNTL(XM,STD,SK,XMR)
C
      COMMON/QUANT/Q10,Q25,Q50,Q100,Q200,Q500
      DIMENSION XK(15,111),CDF(15),RTPK(15),RTLF(15),K(15),Q(15),
     1XJ(111),X1(111),X2(111),X3(111),X4(111),X5(111),X6(111),
     1X7(111),X8(111),X9(111),X10(111),X11(111),X12(111),
     1X13(111),X14(111),X15(111)
      REAL K
      DATA CDF/.005,.01,.02,.04,.1,.2,.5,.8,.9,.96,.98,.99,.995,.998,
     1.999/
      DATA X1/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.606,-.625,-.6452,
     3-.6667,-.6896,-.7143,-.7407,-.7691,-.7997,-.8328,-.8686,-.9074,
     4-.9495,-.995,-1.0443,-1.0975,-1.1548,-1.2162,-1.2817,-1.3511,
     5-1.4244,-1.5011,-1.5811,-1.6639,-1.7492,-1.8366,-1.9258,-2.0164,
     6-2.10825,-2.2009,-2.2942,-2.388,-2.4819,-2.5758,
     70.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,
     80.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,25*0./
      DATA X2/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.6061,-.625,-.6451,
     1-0.6666,-0.6896,-.7145,-0.7405,-0.7688,-0.7992,-0.832,-0.8672,
     1-0.9052,-0.9461,-0.99,-1.037,-1.0871,-1.1404,-1.1968,-1.2561,
     2-1.3182,-1.3827,-1.4494,-1.5181,-1.5884,-1.66,-1.7327,-1.8062,
     3-1.8803,-1.9547,-2.0293,-2.1039,-2.1784,-2.2526,-2.3264,55*0./
      DATA X3/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44444,-.45455,-.46512,-.4762,-.4878,-.5,-.5128,
     2-.5263,-.5405,-.55556,-.5714,-.5882,-.6061,-.625,-.6451,
     1-0.6665,-0.6894,-.7138,-0.7399,-0.7678,-0.79765,-0.8296,-0.8637,
     1-0.9001,-0.9388,-0.9798,-1.0231,-1.0686,-1.1163,-1.1658,-1.2172,
     1-1.27,-1.3241,-1.3793,-1.4353,-1.4919,-1.5489,-1.606,-1.6633,
     2-1.7203,-1.7772,-1.8336,-1.8896,-1.945,-1.9997,-2.0538,55*0./
      DATA X5/-.36364,-.3704,-.3774,-.38462,-.3922,-.4,-.4082,-.4167,
     1-.4255,-.4348,-.44443,-.45452,-.4651,-.4761,-.4877,-.4999,
     2-.5126,-.526,-.5401,-.5548,-.57035,-.5867,-.6038,-.62175,-.6406,
     1-0.6602,-0.6808,-.7021,-0.7242,-0.7471,-0.7706,-0.7947,
     1-0.8193,-0.8442,-0.8694,-0.8946,-0.9199,-0.945,-0.9698,-0.9942,
     2-1.0181,-1.0414,-1.0641,-1.0861,-1.1073,-1.1276,-1.1471,-1.1657,
     3-1.1835,-1.2003,-1.2162,-1.2311,-1.2452,-1.2582,-1.2704,-1.2816,
     455*0./
      DATA X6/-.3636,-.3704,-.37734,-.38458,-.39211,-.39993,-.40806,
     1-.4165,-.4253,-.4345,-.444,-.454,-.4643,-.475,-.4862,-.4978,
     2-.5099,-.5224,-.5353,-.5487,-.5624,-.5765,-.591,-.6057,-.6206,
     1-0.6357,-0.6509,-.666,-0.6811,-0.696,-0.7107,-0.725,-0.7388,
     1-0.7521,-0.7648,-0.7769,-0.7882,-0.7987,-0.8084,-0.8172,-0.8252,
     2-.8322,-.8384,-.8437,-.8481,-.8516,-.8543,-.8561,-.857,-.8572,
     3-.8565,-.8551,-.8529,-.8499,-.8461,-.8416,55*0./
      DATA X7/-.3546,-.3596,-.3645,-.3695,-.3743,-.379,-.3836,-.388,
     1-.3922,-.3962,-.3999,-.4032,-.4062,-.4088,-.411,-.4127,-.4138,
     2-.4144,-.4144,-.4138,-.4125,-.4106,-.4079,-.4045,-.4004,
     1-.3955,-.3899,-.3835,-.3764,-.3685,-.3599,-.3506,-.3406,
     1-.33,-.3187,-.3069,-.2944,-.2815,-.2681,-.2542,-.24,-.2254,
     2-.2104,-.1952,-.1797,-.164,-.1481,-.132,-.1158,-.0995,-.083,
     3-.0665,-.0499,-.0333,-.0166,0.,55*0./
      DATA X8/-.0103,.00243,.0156,.0293,.0434,.058,.073,.0885,.1044,
     1.1207,.1374,.1545,.1719,.1897,.2078,.2262,.2448,.2638,.2829,.3022,
     2.3217,.3413,.361,.3808,.4006,
     1.4204,.4402,.4598,.4793,.4987,.5179,.5368,.5555,.5738,
     1.5918,.6094,.6266,.6434,.6596,.6753,.6905,.7051,.7192,.7326,
```

```
2.7454,.7575,.769,.7799,.79,.7995,.8083,.8164,.8238,.8304,.8364,
3.8416,55*0./
 DATA X9/.6912,.712,.7328,.7536,.7746,.7955,.8164,.8373,.8582,
1.879,.8996,.9202,.9406,.9609,.981,1.0008,1.0204,1.0397,1.0586,
21.0773,1.0955,1.1134,1.1308,1.1477,1.1642,1.1801,1.1954,
11.2101,1.2242,1.2377,1.2504,1.2624,1.2737,1.2841,1.2938,
11.3026,1.3105,1.3176,1.3238,1.329,1.3333,1.3367,1.339,1.3405,
21.3409,1.3404,1.3389,1.3364,1.3329,1.3285,1.3231,1.3167,1.3094,
31.3011,1.2918,1.2816,55*0./
 DATA X10/2.0474,2.0637,2.0795,2.0949,2.1099,2.1243,2.1383,2.1517,
12.1647,2.177,2.1887,2.1999,2.2104,2.2202,2.2294,2.2379,2.2456,
22.2525,2.2587,2.2641,2.2686,2.2723,2.2751,2.2769,2.2779,2.2778,
12.27676,2.2747,2.2716,2.2674,2.2622,2.2558,2.2483,2.2397,
12.2299,2.2189,2.2067,2.1933,2.1787,2.1629,2.1459,2.1277,2.1082,
22.0876,2.0657,2.0427,2.0185,1.9931,1.9666,1.939,1.9102,1.8804,
31.8495,1.8176,1.7846,1.7507,1.7158,1.68,1.6433,1.6057,1.5674,
41.5283,1.4885,1.4481,1.4072,1.3658,1.3241,1.2823,1.2403,1.1984,
51.1568,1.1157,1.0751,1.0354,.9967,.9592,.923,.8881,.8549,.8232,
6.7931,.7646,.7377,.7123,.6884,.6659,
6.6447,.6247,.6059,.5881,.5714,.5555,.5405,.5263,.5128,.5,.4878,
7.4762,.4651,.4546,.4444,.4348,.4255,.4167,.4082,.4,.3922,.3846,
8.3774,.3704,.3636/
 DATA X11/3.2838,3.2884,3.2924,3.2957,3.2982,3.30007,3.3012,
13.3015,3.301,3.2998,3.2977,3.2947,3.2909,3.2862,3.2806,3.274,
23.2665,3.258,3.2485,3.238,3.2264,3.2138,3.2,3.1851,3.1691,
33.1519,3.1336,3.114,3.0932,3.0712,3.0479,3.0233,2.9974,2.9703,
42.9418,2.912,2.8809,2.8485,2.8147,2.7796,2.7433,2.7056,2.6666,
52.6263,2.5848,2.5421,2.4981,2.453,2.4067,2.3593,2.3108,2.2613,
62.2108,2.1594,2.107,2.0538,55*0./
 DATA X12/4.6402,4.6285,4.6159,4.6025,4.5882,4.573,4.5569,4.5399,
14.5219,4.503,4.483,4.4621,4.4401,4.4171,4.393,4.3678,4.3415,4.314,
24.2855,4.2557,4.2247,4.1926,4.1592,4.1245,4.0886,
34.0514,4.0129,3.973,3.9318,3.8893,3.8454,3.8001,3.7535,3.7054,
43.656,3.6052,3.553,3.4994,3.4444,3.388,3.3304,3.2713,3.211,
53.1494,3.0866,3.0226,2.9574,2.891,2.8236,2.7551,2.6857,2.6154,
62.5442,2.4723,2.3996,2.3264,55*0./
 DATA X13/6.08307,6.0517,6.0193,5.986,5.9517,5.9164,5.88,5.8427,
15.8042,5.7646,5.724,5.6822,5.6393,5.5953,5.5501,5.5036,5.456,
25.4071,5.357,5.3056,5.2529,5.1989,5.1436,5.087,5.029,
34.9696,4.9088,4.8467,4.7831,4.7182,4.6518,4.5839,4.5147,4.444,
44.3719,4.2983,4.2234,4.147,4.0693,3.9902,3.9097,3.828,3.745,
53.6607,3.5753,3.4887,3.4011,3.3124,3.2228,3.1323,3.041,2.949,
62.8564,2.7632,2.6697,2.5758,55*0./
 DATA X14/8.0869,8.0259,7.9639,7.9008,7.8366,7.7712,7.7048,
17.6372,7.5684,7.4985,7.4273,7.355,7.2814,7.2065,7.1304,7.053,
26.9744,6.8944,6.813,6.7303,6.6463,6.5608,6.474,6.3858,6.2961,
16.20506,6.1125,6.0186,5.9232,5.8263,5.728,5.6282,5.5269,
15.4243,5.3201,5.2146,5.1077,4.9994,4.8897,4.7788,4.6665,4.553,
24.4384,4.3226,4.2058,4.088,3.9693,3.8498,3.7296,3.6087,3.4874,
33.3657,3.2437,3.1217,2.9998,2.8782,2.7571,2.6367,2.5174,
42.3994,2.2831,2.1688,2.057,1.9481,1.8424,1.7406,1.6431,1.5502,
51.4623,1.3798,1.3028,1.2313,1.1653,1.1047,1.049,.998,.9513,.9085,
6.8693,.8332,.7999,.7692,.7407,.7143,.6896,.6667,25*0./
 DATA X15/9.6577,9.5723,9.4859,9.3983,9.3095,9.2196,9.1285,9.0362,
18.9427,8.848,8.752,8.6548,8.5563,8.4565,8.3553,8.2529,8.1491,
28.044,7.9374,7.8295,7.7202,7.6095,7.4974,7.3838,7.2688,
17.1524,7.0344,6.9151,6.7942,6.6719,6.5481,6.4229,6.2963,
26.1682,6.0387,5.9078,5.7755,5.6419,5.507,5.3709,5.2335,5.0951,
34.9555,4.8149,4.6734,4.5311,4.3881,4.2444,4.1002,3.9557,3.8109,
43.6661,3.5214,3.377,3.2332,3.0902,2.9483,2.8079,2.6692,2.5326,
```

```
     52.3987,2.2678,2.1405,2.0174,1.8989,1.7857,1.6783,1.577,1.4822,
     61.3941,1.3128,1.2381,1.1697,1.1074,1.0507,.999,.9519,.9089,.8695,
     7.8333,.8,.7692,.7407,.7143,.6897,.6667,25*0./                          72
      J=111
      DO 61 I=1,25
      X14(J)=-X1(I)
      X15(J)=-X1(I)
61    J=J-1
      J=111
      DO 62 I=1,111
      X4(J)=-X10(I)
62    J=J-1
      J=111
      DO 63 I=1,55
      X1(J)=-X13(I)
      X2(J)=-X12(I)
      X3(J)=-X11(I)
      X5(J)=-X9(I)
      X6(J)=-X8(I)
      X7(J)=-X7(I)
      X8(J)=-X6(I)
      X9(J)=-X5(I)
      X11(J)=-X3(I)
      X12(J)=-X2(I)
      X13(J)=-X1(I)
63    J=J-1
      DO 1 J =1,111
      XK(1,J)=X1(J)
      XK(2,J)=X2(J)
      XK(3,J)=X3(J)
      XK(4,J)=X4(J)
      XK(5,J)=X5(J)
      XK(6,J)=X6(J)
      XK(7,J)=X7(J)
      XK(8,J)=X8(J)
      XK(9,J)=X9(J)
      XK(10,J)=X10(J)
      XK(11,J)=X11(J)
      XK(12,J)=X12(J)
      XK(13,J)=X13(J)
      XK(14,J)=X14(J)
      XK(15,J)=X15(J)
 1    CONTINUE
      DO 65 I=1,15
      RTLF(I)=1./CDF(I)
65    RTPK(I)=1./(1.-CDF(I))
      RTPK(15)=1000.
      RTPK(14)=500.
      RTPK(13)=200.
      RTPK(12)=100.
      J=1
301   W=J
      XJ(J)=5.6-W/10.0
      IF(XJ(J)-SK)303,303,302
302   J=J+1
      GO TO 301
303   DO 304 I=9,14
      VK =((SK-XJ(J))*(XK(I,J-1)-XK(I,J)))/(XJ(J-1)-XJ(J))+XK(I,J)
      K(I)=EXP(XM+VK*STD)
304   CONTINUE
```

```
      DO 305 I=9,14
 305  Q(I)=K(I)*XMR
      Q10=Q(9)
      Q25=Q(10)
      Q50=Q(11)
      Q100=Q(12)
      Q200=Q(13)
      Q500=Q(14)
C
C     WRITE(6,310)
C310  FORMAT(/' *** LOG PEARSON VARIATE ESTIMATES BY MXM1 METHOD ***'//
C    19X,'CDF',2X,'T(FOR LOS)',2X,'T(FOR PKS)',5X,'VARIATE',//)
C     DO 315 I=1,15
C     WRITE(6,320) CDF(I),RTLF(I),RTPK(I),Q(I)
C315  CONTINUE
C     WRITE(18,330)Q(7),Q(8),Q(9),Q(11),Q(12),Q(13),Q(14)
C330  FORMAT(3X,'MOMIX',1X,7(2X,F7.1))
C320  FORMAT(3F12.3,F12.2)
C     WRITE(6,325)
C325  FORMAT(' NOTE: T=RETURN PERIOD(YRS), LOS=MINIMUM VALUES LIKE LOW F
C    1LOWS, PKS=MAXIMUM VALUES LIKE FLOOD FLOWS')
      RETURN
      END
$ENTRY
0.3  1.0   0.008640   1134.579    -9.844939
0.3  3.0   0.150978   2.681889    -0.438946
0.3  5.0   0.212816   1.135427    -0.271700
0.5  1.0  -0.118320   19.822690    2.216713
0.5  3.0   0.127683   10.303120   -1.407434
0.5  5.0   0.205678   3.215257    -0.740366
0.7  1.0  -0.402431   4.640732     1.569527
0.7  3.0   0.059798   98.38009    -6.066213
0.7  5.0   0.168073   9.569410    -1.760870
//GO.FT10F001 DD DSN=CEAROR.LP3.PPQN,DISP=SHR
$$
//
```

```
//PROJECT JOB (1304,59634,2,20),'ARORA',MSGCLASS=S,CLASS=Q
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=D
//     EXEC WATFIV,REGION.GO=4000K,TIME.GO=99
$JOB           TIME=4500
C
C*************************************************************************
C                  LOG PEARSON TYPE 3 DISTRIBUTION
C-------------------------------------------------------------------------
C    PROGRAM TO CALCULATE POPULATION PARAMETERS FOR GIVEN MEAN (=1 FOR
C    DIMENSIONLESS VARIATE), COEFF. OF VARIATION, AND SKEW COEFF. OF LP3
C*************************************************************************
C
      COMMON/ALL/BTAB(298),ALPTAB(298)
      COMMON/MMPAR/A,B,C
C
      READ(9,*)(BTAB(I),ALPTAB(I),I=1,298)
C
      DELCV=0.05
      DELSK=0.1
      CVV=0.30
      DO 10 I=1,10
      SKEW=1.00
      WRITE(6,*)'  '
      WRITE(6,*)'  '
      WRITE(6,*)'     COEFF. OF VARIATION = ',CVV
      WRITE(6,*)'=============================================================
     1============================================================='
      DO 20 J=1,41
      CALL MOMENT(CVV,SKEW)
      CALL POLATE
      CALL MMDIR
C     WRITE(6,*)'  '
      WRITE(6,1)SKEW,A,B,C
    1 FORMAT(2X,'SKEWNESS COEFFICIENT = ',F6.2,'  |   PARAMETERS :   ',
     +3F14.6)
      SKEW=SKEW+DELSK
   20 CONTINUE
      CVV=CVV+DELCV
   10 CONTINUE
      STOP
      END
C
C           ********   END OF MAIN SEGMENT   ********
C
C*************************************************************************
C     SUBROUTINE MOMENT COMPUTES THE FIRST THREE MOMENTS OF THE SAMPLE
C     AND CALCULATES B AS A FUNCTION OF THE THREE MOMENTS
C*************************************************************************
      SUBROUTINE MOMENT(CV,SKW)
      COMMON/LNMMNT/AL1,AL2,AL3
      COMMON/BVAL/B
      REAL L1,L2,L3
      SIGMA=CV
```

```
      SKEW=SKW
      SIGMA2=SIGMA*SIGMA
      SIGMA3=SIGMA2*SIGMA
      L1=1.0
      L2=1.0+SIGMA2
      L3=1.0+3.0*SIGMA2+SIGMA3*SKEW
C
      AL1=ALOG(L1)
      AL2=ALOG(L2)
      AL3=ALOG(L3)
      B=(AL3-3.0*AL1)/(AL2-2.0*AL1)
C     WRITE(6,*)' FACTOR B USED IN MOMDIR = ',B
      RETURN
      END
C
C*************************************************************************
C     SUBROUTINE POLATE INTERPOLATES FOR DIRECT METHOD OF MOMENTS
C     USING BTAB(.),ALPTAB(.)
C*************************************************************************
      SUBROUTINE POLATE
      COMMON/BVAL/B
      COMMON/EST/ALPEST
      COMMON/ALL/BTAB(298),ALPTAB(298)
      DO 10 I= 1,298
      IF((B .LT. 2.04622).OR. (B .GT.8.56194))GO TO 12
      IF((B.GE.BTAB(I)).AND. (B .LE. BTAB(I+1)))GO TO 50
 10   CONTINUE
 50   DELALP= (ALPTAB(I)-ALPTAB(I+1))/(BTAB(I)-BTAB(I+1))
     1*(B-BTAB(I+1))
      ALPEST= DELALP+ALPTAB(I+1)
      RETURN
 12   WRITE(6,3)
  3   FORMAT(1X,'NO DIRECT MOMENT SOLUTION POSSIBLE')
      RETURN
      END
C
C*************************************************************************
C     SUBROUTINE TO CALCULATE THE PARAMETERS BY METHOD OF MOMENTS
C*************************************************************************
      SUBROUTINE MMDIR
      COMMON/EST/ALPEST
      COMMON/MMPAR/ALPHA,BETA,GAMMA
      COMMON/LNMMNT/AL1,AL2,AL3
C
      ALPHA=ALPEST
      A1=ALOG(1.0-ALPHA)
      A2=ALOG(1.0-2.0*ALPHA)
      BETA=(AL2-2.0*AL1)/(2.0*A1-A2)
      GAMMA=AL1+BETA*A1
      RETURN
      END
$ENTRY
1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5 6.0
//GO.FT09F001 DD DSN=CEAROR.LPT.DATA1,DISP=SHR
$$
//
0.35 0.40 0.45 0.50 0.55 0.60 0.65 0.70
//GO.FT08F001 DD DSN=CEAROR.TCEV.****,DISP=SHR
```

```
//PROJECT JOB (1304,59634,2,20),'ARORA',MSGCLASS=S,CLASS=Q
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=D
//    EXEC WATFIV,REGION.GO=4000K,TIME.GO=99
$JOB          TIME=4500
      DIMENSION AA(50),IBB(50)
C
C*************************************************************************
C                  LOG PEARSON TYPE 3 DISTRIBUTION
C-------------------------------------------------------------------------
C    PROGRAM TO COMPUTE POPULATION COEFF. OF VARIATION(CV), SKEW COEFF.,
C    AND PARAMETER C FOR GIVEN A AND B( = AN INTEGER VALUE)
C        P.S. : THE VALUE OF B IS THE NEAREST INTEGER VALUE PICKED FROM
C        ---    THE OUTPUT OF 'PPLATN' FOR SOME CV-SKEW COMBINATIONS
C*************************************************************************
C
      NCASES=3
      DO 10 I=1,NCASES
      READ(5,*)AA(I),IBB(I)
  10  CONTINUE
      WRITE(6,*)'       A          B          C        CV          SKEW'
      DO 20 I=1,NCASES
      B=FLOAT(IBB(I))
      A=AA(I)
      AL1=ALOG(1.0-A)
      AL2=ALOG(1.0-2.*A)
      AL3=ALOG(1.0-3.*A)
      C=B*AL1
      T2=B*(2.0*AL1-AL2)
      VRNCE=EXP(T2)-1.0
      SIGMA=SQRT(VRNCE)
      T3=B*(3.*AL1-AL3)
      SKEW=(EXP(T3)-(1.0+3.*VRNCE))/(VRNCE*SIGMA)
      WRITE(6,*)A,B,C,SIGMA,SKEW
  20  CONTINUE
      STOP
      END
$ENTRY
0.1269 4
0.1133 10
0.0932 21
$$
//
//GO.FT08F001 DD DSN=CEAROR.TCEV.****,DISP=SHR
```

```
//PROJECT JOB (1304,59634,2,20),'ARORA',MSGCLASS=S,CLASS=Q
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=D
//     EXEC WATFIV,REGION.GO=4000K,TIME.GO=99
$JOB            TIME=4500
C*********************************************************************
C                  LOG PEARSON TYPE 3 DISTRIBUTION
C---------------------------------------------------------------------
C      GENERATES P(.) AND A(.) TO BE USED IN METHOD OF MIXED MOMENTS.
C      THESE VALUES ARE READ IN M/PROG OF LP3.SIMU(M5)..MIX
C  NOTE : P CALCULATED IN HERE IS -VE OF ACTUAL ONE. THE SIGN IS
C           ADJUSTED IN LP3.SIMU(M5)
C*********************************************************************
       DOUBLE PRECISION A1,A2,A,DP
       AA=0.499
       DEL=0.0001
C
       A=AA
       DO 10 I=1,6
       A1=DLOG(1.D0-A)
       A2=DLOG(1.D0-2.D0*A)
       DP=(-2.D0*A1+A2)/(A+A1)
       P=DP
       WRITE(6,*)I,DP,AA,A
       WRITE(10,*)P,AA
       AA=AA+DEL
       A=AA
  10   CONTINUE
       STOP
       END
$ENTRY
//GO.FT10F001 DD DSN=CEAROR.LPT.DATA2,DISP=SHR
$$
//
//GO.FT09F001 DD DSN=CEAROR.LPT.DATA1,DISP=SHR
```

```
//PROJECT JOB (1304,59634,2,20),'ARORA',MSGCLASS=S,CLASS=Q
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=D
//     EXEC WATFIV,REGION.GO=4000K,TIME.GO=99
$JOB            TIME=4500
C*******************************************************************
C                LOG PEARSON TYPE 3 DISTRIBUTION
C-------------------------------------------------------------------
C      GENERATES BTAB(.) AND ALPTAB(.) TO BE USED IN DIRECT METHOD OF
C      MOMENTS.  THESE VALUES ARE READ IN M/PROG OF LP3.SIMU(M1)
C*******************************************************************
       DIMENSION BTAB(500),ALPTAB(500)
       DOUBLE PRECISION A1,A2,A3,B,A,DEL
C
       A=-1000.0D0
       DEL=-200.0D0
       DO 10 I=1,20
       ALPTAB(I)=A
       A1=DLOG(1.D0-A)
       A2=DLOG(1.D0-2.D0*A)
       A3=DLOG(1.D0-3.D0*A)
       B=(3.D0*A1-A3)/(2.D0*A1-A2)
       BTAB(I)=B
       WRITE(6,*)BTAB(I),ALPTAB(I),A
       WRITE(10,*)BTAB(I),ALPTAB(I)
       A=A+DEL
  10   CONTINUE
       STOP
       END
$ENTRY
//GO.FT10F001 DD DSN=CEAROR.LPT.DATA2,DISP=SHR
$$
//
//GO.FT09F001 DD DSN=CEAROR.LPT.DATA,DISP=SHR
//GO.FT08F001 DD DSN=CEAROR.TCEV.****,DISP=SHR
```

```
//PROJEC1 JOB (1304,59634,2,20),'ARORA',MSGCLASS=S,CLASS=Q                00010000
/*ROUTE PRINT CEBA                                                        00020000
/*JOBPARM SHIFT=N
//     EXEC WATFIV,REGION.GO=4000K,TIME.GO=99
$JOB           TIME=4500
C
C*************************************************************************
C                  LOG PEARSON TYPE 3 DISTRIBUTION
C-------------------------------------------------------------------------
C      PROGRAM TO PREPARE  P(.) AND A(.) TABLE FOR THE METHOD OF
C                          MIXED MOMENTS (MIX).
C*************************************************************************
C
      DIMENSION A(150),P(150)
      DO 5 I=1,141
      READ(9,*)P(I),A(I)
C     P(I)=-P(I)
  5   CONTINUE
C
      DO 10 I=1,47
      J=47+I
      K=94+I
      WRITE(6,1)A(I),P(I),A(J),P(J),A(K),P(K)
      WRITE(10,1)A(I),P(I),A(J),P(J),A(K),P(K)
   1  FORMAT(16X,F11.5,F10.5,'    *',F11.5,F10.5,'    *',F11.5,F10.5)
  10  CONTINUE
      STOP
      END
$ENTRY
//GO.FT09F001 DD DSN=CEAROR.LPT.DATA1,DISP=SHR
//GO.FT10F001 DD DSN=CEAROR.LPT.TMMD,DISP=SHR
$$
//
//GO.FT09F001 DD DSN=CEAROR.LPT.DATA1,DISP=SHR
```

```
//PROJE9 JOB (1304,59634,30,20),'ARORA',MSGCLASS=S
/*ROUTE PRINT CEBA
//     EXEC FORTVCLG,REGION.GO=4000K,TIME.GO=200
/*JOBPARM SHIFT=N
//FORT.SYSIN DD *
C
C*************************************************************************
C          EXTREME VALUE TYPE 1 (GUMBEL) DISTRIBUTION
C-------------------------------------------------------------------------
C   1) CACULATES THE PARAMETERS AND QUANTILE ESTIMATES OF THE GUMBEL'S
C        DISTRIBUTION USING SEVERAL(7) ESTIMATION METHODS.
C   2) BOTH RANDOM AND SERIALLY CORRELATED SAMPLES ARE CONSIDERED
C   3) CALCULATES THE STATISTICAL PROPERTIES (BIAS, STD. DEVIATION,
C      AND MEAN SQUARE ERROR) OF THE ESTIMATORS (MONTE-CARLO SIMULATION)
C*************************************************************************
C
      DOUBLE PRECISION DSEED
      EXTERNAL FF
      INTEGER SSIZE(15)
      REAL MDNOR
      REAL MA(10),MB(10),MA2(10),MB2(10),MQX(10,15),MSEA(10),MSEB(10)
      REAL MQX2(10,15),MSEQX(10,15)
      DIMENSION TITLE(80),G(100),Q(30),ID(10,10),EFFA(10),EFFB(10)
      DIMENSION SA(10),SB(10),SA2(10),SB2(10),RA(1200),XN(7)
      DIMENSION BA(10),BB(10),STDA(10),STDB(10),ALPHA(2)
      DIMENSION QXP(15),BIAQX(10,15),FD(15),SQX(10,15),QX(10,15)
      DIMENSION EFFQX(10,15),SQX2(10,15),STDQX(10,15)
      DIMENSION Z(1200)
      COMMON/PARA/X(1200),N
      COMMON/RESUL/A(7),B(7)
      COMMON/STAT/XAVG,SIGMA,CS,TS
      COMMON/TEST/C(7),CRIT
      COMMON/ENTR/ENT(7)
      COMMON/CLASS/F(42)
      COMMON/CHIS/R(7)
      COMMON/INTER/CIUP(6),CILO(6),T(6)
      COMMON/PEST/BIAS(7)
      COMMON/ERR/ERROR(7),ERRO(7)
C-------------------------------------------------------------------------
C-------------GUMBEL POPULATION PARAMETERS :APOP AND BPOP :-----------
              APOP=1.00
              BPOP=0.0
              WRITE(6,543)APOP,BPOP
 543          FORMAT(/,5X,'GUMBEL PARAMETERS : ',2X,'A = ',F6.3,/,
     1        '                              B = ',F7.3,/)
C-------------------------------------------------------------------------
      READ(5,77)(XN(I),I=1,7)
  77  FORMAT(7(A3))
      READ(5,*)(FD(IQ),IQ=1,13)
      DO 444 IQ=1,13
 444  QXP(IQ)=BPOP-(ALOG(-ALOG(FD(IQ))))/APOP
C     WRITE(6,*)(QXP(IK),IK=1,13)
C-------------------------------------------------------------------------
C     WRITE(8,98)
```

```
C     WRITE(9,66)(FD(I),I=1,13)
C     WRITE(9,66)(QXP(I),I=1,13)
C     WRITE(10,66)(FD(I),I=1,13)
C     WRITE(10,66)(QXP(I),I=1,13)
C     WRITE(11,66)(FD(I),I=1,13)
C     WRITE(11,66)(QXP(I),I=1,13)
C-----------------------------------------------------------------------
      DSEED=234567.D0
      WRITE(6,191)DSEED
 191  FORMAT(/,2X,'INITIAL SEED = ',D23.16,/)
C-    ##################################################
      NCASE=1
C-    ##################################################
      READ(5,*)(SSIZE(I),I=1,NCASE)
C-----------------------------------------------------------------------
      DO 51 IS=1,NCASE
      IC=0
      NR=SSIZE(IS)
      N=NR
      DSEED=234567.D0
      IF(NR.GT.200)GO TO 789
      NS=50000
      GO TO 788
 789  NS=10
 788  WRITE(6,155)NS
 155  FORMAT(130('*'),//,2X,'NO. OF SAMPLES = ',I5,/)
          DO 11 IJ=1,7
          SA(IJ)=0.0
          SB(IJ)=0.0
          SA2(IJ)=0.0
          SB2(IJ)=0.0
  11      CONTINUE
      DO 446 IP=1,7
      DO 446 IQ=1,13
      SQX(IP,IQ)=0.0
 446  SQX2(IP,IQ)=0.0
C-----------------------------------------------------------------------
      Z0=0.0
      RHO=0.5
C
      DO 50 I=1,NS
C
      CALL GGUBS(DSEED,NR,RA)
C
           DO 111 J =1,NR
           RAA=RA(J)
 111       X(J)=BPOP-(ALOG(-ALOG(RAA)))/APOP
C
C-------------------------------------------------------------
C -------SERIALLY CORRELATED GUMBEL NUMBERS NOW. -------------
C     CALL GGNML(DSEED,NR,RA)
C
C     DO 111 J=1,NR
C     Z(J)=RHO*Z0+((1.-RHO**2.)**0.5)*RA(J)
C     Z1=Z(J)
C     Z0=Z(J)
C     CALL MDNOR(Z1,PROB)
C     X(J)=BPOP-(ALOG(-ALOG(PROB)))/APOP
C111  CONTINUE
C
```

```
C-------------------------------------------------------------
      CALL VSRTA(X,NR)
C     WRITE(6,*)(X(J),J=1,NR)
C-------------------------------------------------------------
C     CALL PARAM
C     WRITE(6,*)N,XAVG,SIGMA,CS,TS
C.....................................
C
      CALL MM(1)
C
C.....................................
C     IF(FLOAT(N).GT.30.0)GO TO 808
C     A(1)=A11*(1.0-(0.36/(FLOAT(N)**0.88)))
C     GO TO 809
C808  A(1)=A11*(1.0-(0.27/(FLOAT(N)**0.80)))
C------***********----------**************-------------------
C809  CONTINUE
      DO 501 IKL=2,7
      A(IKL)=A(1)
      B(IKL)=B(1)
 501  CONTINUE
C------***********----------**************-------------------
C         WRITE(6,*)A(1),B(1)
C     GO TO 234
C---------------------------------------------------------------------
C      THE FOLLOWING SUBROUTINE ZREAL2 (IMSL) IS USED TO CROSS-CHECK
C     THE MLE PARAMETER ESTIMATES FROM SUBROUTINE MLE(IS).
C.....................................................
C     EPS=1.0E-5
C     EPS2=1.0E-5
C     ETA=1.0E-2
C     NSIG=4
C     ITMAX=100
C     N1=1
C     ALPHA(1)=1.0
C     CALL ZREAL2(FF,EPS,EPS2,ETA,NSIG,N1,ALPHA,ITMAX,IER)
C     A(2)=ALPHA(1)
C     SU=0.0
C     DO 1111 II=1,N
C     TES=EXP(-A(2)*X(II))
C1111 SU=SU+TES
C     B(2)=(ALOG(FLOAT(N))-ALOG(SU))/A(2)
C         WRITE(6,*)ALP,BLP
C---------------------------------------------------------------------
C     EPS=1.0E-5
C     NSIG=5
C     AI=0.3*A(1)
C     BI=4.0*A(1)
C     MAXFN=100
C     CALL ZBRENT(FF,EPS,NSIG,AI,BI,MAXFN,IER)
C     A(2)=BI
C     SM=0.0
C     DO 1111 II=1,N
C     DEL=EXP(-A(2)*X(II))
C1111 SM=SM+DEL
C     B(2)=(ALOG(FLOAT(N))-ALOG(SM))/A(2)
C---------------------------------------------------------------------
C234  CALL MLE(2)
C         WRITE(6,*)A(2),B(2)
C     CALL PWM(3)
```

```
C         WRITE(6,*)A(3),B(3)
C     CALL ET(4)
C         WRITE(6,*)A(4),B(4)
C     CALL LEAST(5)
C         WRITE(6,*)A(5),B(5)
C     CALL MMM(6)
C         WRITE(6,*)A(6),B(6)
C     ICOUNT=0
C     CALL IM(7,ICOUNT)
C     WRITE(6,*)ICOUNT,A(7),B(7)
C     IC=ICOUNT+IC
C         WRITE(6,*)IC
C#################################################################
C..... . . . . .QUANTILES NOW. . . . . .................
C     WRITE(6,64)
C 64  FORMAT(1X,'TESTING')
C
      DO 448 IP=1,7
      DO 448 IQ=1,13
      IF((IP.EQ.7).AND.(ICOUNT.EQ.1))GO TO 777
      ALAL=(ALOG(-ALOG(FD(IQ))))
      QX(IP,IQ)=B(IP)-ALAL/A(IP)
      IF(IP.EQ.1)GO TO 781
      GO TO 778
 781  FN=0.35/((FLOAT(N))**0.8589)
      CORR=(0.57721+ALAL)/(A(IP)*(1.0-FN))
      QX(IP,IQ)=QX(IP,IQ)-FN*CORR
      GO TO 778
 777  QX(IP,IQ)=0.0
 778  SQX(IP,IQ)=SQX(IP,IQ)+QX(IP,IQ)
 448  SQX2(IP,IQ)=SQX2(IP,IQ)+QX(IP,IQ)*QX(IP,IQ)
C
C     WRITE(6,65)((QX(IL,JL),JL=1,13),IL=1,7)
C     WRITE(6,65)((SQX(IL,JL),JL=1,13),IL=1,7)
C 65  FORMAT(2X,13(F9.2))
C-------------------------------------------------------------
C     CALL KSTEST
C     CALL SURP
C     CALL CI(ID)
C     CALL CHI(ID)
C     CALL CONINT
C     CALL BIASA
C     CALL RMSE
C     ----------------
          DO 222 K=1,7
          SA(K)=SA(K)+A(K)
          SB(K)=SB(K)+B(K)
          SA2(K)=SA2(K)+A(K)*A(K)
          SB2(K)=SB2(K)+B(K)*B(K)
 222      CONTINUE
 50   CONTINUE
C-----------------------------------------------------------------------
C
          DO 333 K=1,7
C         WRITE(6,*)IC
          IF(K.EQ.7)GO TO 301
          NDIV=NS
          GO TO 302
 301      NDIV=NS-IC
 302      MA(K)=SA(K)/FLOAT(NDIV)
```

```
          MB(K)=SB(K)/FLOAT(NDIV)
          MA2(K)=SA2(K)/FLOAT(NDIV)
          MB2(K)=SB2(K)/FLOAT(NDIV)
 333      CONTINUE
          DO 345 K=1,7
          STDA(K)=(MA2(K)-(MA(K)*MA(K)))**0.5
          STDB(K)=(MB2(K)-(MB(K)*MB(K)))**0.5
          BA(K)=APOP-MA(K)
          BB(K)=BPOP-MB(K)
          MSEA(K)=BA(K)*BA(K)+STDA(K)*STDA(K)
          MSEB(K)=BB(K)*BB(K)+STDB(K)*STDB(K)
 345      CONTINUE
          DO 346 K=1,7
          EFFA(K)=MSEA(2)/MSEA(K)
          EFFB(K)=MSEB(2)/MSEB(K)
 346      CONTINUE
C
      WRITE(6,98)
  98  FORMAT(//,1X,'METHOD  SAMPLE SIZE   BIAS(A)    STD(A)   EFF.(A)
     1BIAS(B)    STD(B)    EFF.(B)',/,1X,80('-'),/)
      DO 350 K=1,7
      WRITE(6,99)XN(K),N,BA(K),STDA(K),EFFA(K),BB(K),STDB(K),EFFB(K)
C     WRITE(8,99)XN(K),N,BA(K),STDA(K),EFFA(K),BB(K),STDB(K),EFFB(K)
 350  CONTINUE
 99   FORMAT(2X,A3,5X,I4,5X,6(F10.3))
C.....................QUANTILES NOW.........................
      DO 452 IP=1,7
      DO 452 IQ=1,13
      IF(IP.EQ.7)GO TO 401
      NDIV=NS
      GO TO 402
401   NDIV=NS-IC
402   MQX(IP,IQ)=SQX(IP,IQ)/FLOAT(NDIV)
      MQX2(IP,IQ)=SQX2(IP,IQ)/FLOAT(NDIV)
      BIAQX(IP,IQ)=QXP(IQ)-MQX(IP,IQ)
      STDQX(IP,IQ)=SQRT(MQX2(IP,IQ)-MQX(IP,IQ)*MQX(IP,IQ))
 452  MSEQX(IP,IQ)=BIAQX(IP,IQ)*BIAQX(IP,IQ)+STDQX(IP,IQ)*STDQX(IP,IQ)
C     WRITE(6,66)((MQX(I,J),J=1,13),I=1,7)
C     WRITE(6,66)((MQX2(I,J),J=1,13),I=1,7)
      DO 453 IP=1,7
      DO 453 IQ=1,13
 453  EFFQX(IP,IQ)=MSEQX(2,IQ)/MSEQX(IP,IQ)
      WRITE(6,68)
      WRITE(6,66)(FD(I),I=1,13)
      WRITE(6,66)(QXP(I),I=1,13)
  66  FORMAT(18X,13(F8.3),/)
      WRITE(6,771)
C     WRITE(9,771)
 771  FORMAT(46X,'BIAS IN QUANTILE ESTIMATES',/,46X,26('-'),/)
      DO 150 I=1,7
 150  WRITE(6,67)XN(I),NR,(BIAQX(I,J),J=1,13)
C150  WRITE(9,67)XN(I),NR,(BIAQX(I,J),J=1,13)
      WRITE(6,68)
      WRITE(6,772)
C     WRITE(10,772)
 772  FORMAT(44X,'STD. DEV. OF QUANTILE ESTIMATES',/,44X,31('-'),/)
      DO 151 I=1,7
 151  WRITE(6,67)XN(I),NR,(STDQX(I,J),J=1,13)
C151  WRITE(10,67)XN(I),NR,(STDQX(I,J),J=1,13)
      WRITE(6,68)
```

```
 68   FORMAT(//)
      WRITE(6,773)
C     WRITE(11,773)
 773  FORMAT(43X,'EFFICIENCY OF QUANTILE ESTIMATES',/,43X,32('-'),/)
      DO 152 I=1,7
 152  WRITE(6,67)XN(I),NR,(EFFQX(I,J),J=1,13)
C152  WRITE(11,67)XN(I),NR,(EFFQX(I,J),J=1,13)
 67   FORMAT(1X,A3,5X,I4,5X,13(F8.3))
      WRITE(6,71)IC
 71   FORMAT(/,5X,'NO.OF BAD SAMPLES FOR ICM ESTIMATION = ',I5,/)
 51   CONTINUE
      STOP
      END
C
C-----------------------------------------------------------------------
C
      SUBROUTINE PARAM
C
C-----------------------------------------------------------------------
C
C     PROGRAM TO CALCULATE THE MEAN  VARIANCE  SKEWNESS AND
C
C     KURTOSIS FOR THE DATA SETS MEMBER NAME(JAN26)
C
C-----------------------------------------------------------------------
C
C
      COMMON/PARA/X(1200),N
      COMMON/RESUL/A(7),B(7)
      COMMON/STAT/XAVG,SIGMA,CS,TS
      DIMENSION SIG(1100),SUM(1100),AUR(1100)
      TOT=0.0
      CS=0.0
      SIGM=0.0
      XTOT=0.0
      TOSIS=0.0
      DO 20 I=1,N
 20   XTOT=XTOT+X(I)
      XAVG=XTOT/FLOAT(N)
      DO 30 I=1,N
      SIG(I)=(X(I)-XAVG)**2
      SUM(I)=(X(I)-XAVG)**3
      AUR(I)=(X(I)-XAVG)**4
      TOT=TOT+SUM(I)
      TOSIS=TOSIS+AUR(I)
 30   SIGM=SIGM+SIG(I)
      SIGMA= SQRT(SIGM/FLOAT(N-1))
      CS=(FLOAT(N)/(FLOAT(N-1)*FLOAT(N-2)))*TOT/(SIGMA**3)
      TS=((FLOAT(N)**2)/(FLOAT(N-1)*FLOAT(N-2)*FLOAT(N-3)))*
     1TOSIS/(SIGMA**4)
      RETURN
      END
C
C
C-----------------------------------------------------------------------
C
C     SUBROUTINE MM CALCULATES THE PARAMETERS OF GUMBEL
C
C           DISTRIBUTION BY METHOD OF MOMENTS
C
```

```
C     N NUMBER OF ANNUAL MAXIMUM EVENTS
C     X SERIES OF EVENTS
C
C-----------------------------------------------------------------------
C
C
      SUBROUTINE MM(IS)
      COMMON/PARA/X(1200),N
      COMMON/RESUL/A(7),B(7)
      REAL M1,M2,M3,K
      DIMENSION T(6)
      DIMENSION XT(6),SX(6)
      T(1)=2.
      T(2)=5.
      T(3)=10.
      T(4)=20.
      T(5)=50.
      T(6)=100.
      XN=N
      AP=0.0
      BP=0.0
      C=0.0
      DO 1 I=1,N
      AP=AP+X(I)
      BP=BP+X(I)*X(I)
      C=C+X(I)*X(I)*X(I)
 1    CONTINUE
      M1=AP/XN
      M2=(BP/XN)-M1*M1
      M2=M2*XN/(XN-1.0)
      M3=(C/XN)+(2.0*M1**3)-((3.0*M1)*(BP/XN))
      SKEW=M3/(M2**1.5)
      A(IS)=3.1415927/(SQRT(M2*6.0))
      B(IS)=M1-0.57721/A(IS)
C
C     AP=0.0
C     BP=0.0
C     DO 2 I=1,N
C     XI=I
C     XN=N
C     Y=-ALOG(-ALOG((XN+1.0-XI)/(XN+1.0)))
C     AP=AP+Y
C     BP=BP+Y**2
C2    CONTINUE
C     YBAR=AP/XN
C     YSTD=SQRT((BP/XN)-YBAR**2)
C     DO 3 J=1,6
C     YM=-ALOG(-ALOG((T(J)-1.0)/T(J)))
C     K=(YM-YBAR)/YSTD
C     XT(J)=M1+K*SQRT(M2)
C     DELTA=1.0+1.139547093*K+1.100000027*K**2
C     SX(J)=SQRT(M2*DELTA/XN)
C3    CONTINUE
C     WRITE (6,20) ALPHA,M1
C     WRITE (6,21) BETA,M2
C     WRITE (6,22) SKEW
C     WRITE (6,25)
C     WRITE (6,14)
C     WRITE (6,15) (XT(J),J=1,6)
C     WRITE (6,16) (SX(J),J=1,6)
```

```
C     WRITE (6,17)
C     WRITE (6,18)
C
C10   FORMAT (I5)
C11   FORMAT (7F10.1)
C12   FORMAT (1H1,/,80A1,//,26X,28H,'TYPE 1 EXTREMAL DISTRI.',/)
C13   FORMAT (31X,17HMETHOD OF MOMENTS,//)
C14   FORMAT (3X,7HT,YEARS,4X,1H2,11X,1H5,10X,2H10,10X,2H20,10X,
C    12H50,9X,3H100,/)
C15   FORMAT (3X,1HX,3X,6E12.5,/,4X,1HT)
C16   FORMAT (3X,1HS,3X,6E12.5,/,4X,1HT,//)
C17   FORMAT (25X,28HMAXIMUM LIKELIHOOD PROCEDURE,//)
C18   FORMAT (21X,5HTRIAL,11X,1HA,11X,4HF(A),/)
C19   FORMAT (22X,I2,8X,E12.5,1X,E12.5)
C20   FORMAT (9X,5HALPHA,5X,E12.5,14X,4HM1 ,6X,E12.5)
C21   FORMAT (9X,5HBETA ,5X,E12.5,14X,4HM2 ,6X,E12.5)
C22   FORMAT (45X,4HSKEW,6X,E12.5,/)
C23   FORMAT (//)
C24   FORMAT (9X,5HBETA ,5X,E12.5,14X,4HM2 ,6X,E12.5,/)
C25   FORMAT (3X,'NOTE - FOR GOOD USE OF THIS DISTRIBUTION SKEW
C    1 SHOULD BE AROUND 1.13',/)
      RETURN
      END
C
      SUBROUTINE MLE(IS)
C     COMPUTES  MAXIMUM LIKELIHOOD ESTIMATE FOR
C     T YEAR EVENTS AND STANDARD ERROR FOR TYPE 1 EXTREMAL DISTRIBUTION
C     INPUT
C     TITLE
C     N NUMBER OF ANNUAL MAXIMUM EVENTS
C     X SERIES OF EVENTS
      COMMON/PARA/X(1200),N
      COMMON/RESUL/A(7),B(7)
      REAL M1,M2,M3,K
      DIMENSION T(6)
      DIMENSION XT(6),SX(6)
      T(1)=2.
      T(2)=5.
      T(3)=10.
      T(4)=20.
      T(5)=50.
      T(6)=100.
      XN=N
      AP=0.0
      BP=0.0
      C=0.0
      DO 1 I=1,N
      AP=AP+X(I)
      BP=BP+X(I)**2
      C=C+X(I)**3
1     CONTINUE
      M1=AP/XN
      M2=(BP/XN)-(AP/XN)**2
      M2=M2*XN/(XN-1.0)
      M3=(C/XN)+(2.0*M1**3)-((3.0*M1)*(BP/XN))
      SKEW=M3/(M2**1.5)
      A(IS)=1.2825/(SQRT(M2))
      B(IS)=M1-0.45*SQRT(M2)
      AP=0.0
      BP=0.0
```

```
      DO 2 I=1,N
      XI=I
      XN=N
      Y=-ALOG(-ALOG((XN+1.0-XI)/(XN+1.0)))
      AP=AP+Y
      BP=BP+Y**2
 2    CONTINUE
      YBAR=AP/XN
      YSTD=SQRT((BP/XN)-YBAR**2)
      DO 3 J=1,6
      YM=-ALOG(-ALOG((T(J)-1.0)/T(J)))
      K=(YM-YBAR)/YSTD
      XT(J)=M1+K*SQRT(M2)
      DELTA=1.0+1.139547093*K+1.100000027*K**2
      SX(J)=SQRT(M2*DELTA/XN)
 3    CONTINUE
C     WRITE (6,13)
C     WRITE (6,20) M1
C     WRITE (6,21) M2
C     WRITE (6,22) SKEW
C     WRITE (6,25)
C     WRITE (6,14)
C     WRITE (6,15) (XT(J),J=1,6)
C     WRITE (6,16) (SX(J),J=1,6)
C     WRITE (6,17)
C     WRITE (6,18)
      ICOUNT=0
      AML=A(IS)
 4    ICOUNT=ICOUNT+1
      AP=1.0/(AML**2)
      BP=M1-1.0/AML
      C=0.0
      D=0.0
      E=0.0
      DO 5 I=1,N
      TEMP=EXP(-AML*X(I))
      C=C+TEMP
      D=D+TEMP*X(I)
      E=E+TEMP*X(I)**2
 5    CONTINUE
      FCN=D-BP*C
      FPN=BP*D-E-AP*C
      AS=AML-(FCN/FPN)
C     WRITE (6,19) ICOUNT,AS,FCN
      DELTA=ABS(0.0000001*AS)
      IF (ABS(AS-AML).LT.DELTA) GO TO 6
      IF (ICOUNT.GT.25) GO TO 6
      AML=AS
      GO TO 4
 6    CONTINUE
      A(IS)=AS
      B(IS)=(1.0/A(IS))*ALOG(XN/C)
      M2=1.2825/A(IS)
      M1=BETA+0.45*M2
      M2=M2**2
      DO 7 J=1,6
      YM=-ALOG(-ALOG(1.0-1.0/T(J)))
      XT(J)=BETA+YM/A(IS)
      SX(J)=SQRT((1.1086+0.5140*YM+0.6079*YM**2)/(XN*A(IS)**2))
 7    CONTINUE
```

```
C     WRITE (6,23)
C     WRITE (6,20) ALPHA,M1
C     WRITE (6,24) BETA,M2
C     WRITE (6,14)
C     WRITE (6,15) (XT(J),J=1,6)
C     WRITE (6,16) (SX(J),J=1,6)
      RETURN
C
C10   FORMAT (I5)
C14   FORMAT (3X,7HT,YEARS,4X,1H2,11X,1H5,10X,2H10,10X,2H20,10X,
C    12H50,9X,3H100,/)
C15   FORMAT (3X,1HX,3X,6E12.5,/,4X,1HT)
C16   FORMAT (3X,1HS,3X,6E12.5,/,4X,1HT,//)
C17   FORMAT (25X,28HMAXIMUM LIKELIHOOD PROCEDURE,//)
C18   FORMAT (21X,5HTRIAL,11X,1HA,11X,4HF(A),/)
C19   FORMAT (22X,I2,8X,E12.5,1X,E12.5)
C20   FORMAT (9X,5HALPHA,5X,E12.5,14X,4HM1 ,6X,E12.5)
C21   FORMAT (9X,5HBETA ,5X,E12.5,14X,4HM2 ,6X,E12.5)
C22   FORMAT (45X,4HSKEW,6X,E12.5,/)
C23   FORMAT (//)
C24   FORMAT (9X,5HBETA ,5X,E12.5,14X,4HM2 ,6X,E12.5,/)
C25   FORMAT (3X,'NOTE - FOR GOOD USE OF THIS DISTRIBUTION SKEW
C    1 SHOULD BE AROUND 1.13',/)
      END
C
C
C*************************************************************************
      SUBROUTINE PWM(IS)
      COMMON/PARA/X(1200),N
      COMMON/RESUL/A(7),B(7)
C*************************************************************************
C
C     THIS PROGRAM CALCULATES THE PARAMETERS OF EV1 DISTRIBUTION
C
C     BY USING METHOD OF PROBABILITY WEIGHTED MOMENTS
C
C*************************************************************************
C
C                   PARAMETERS ARE A AND B
C
C*************************************************************************
C
C                    START OF MAIN PROGRAM
C
C*************************************************************************
      REAL M0,M1
      SUM=0.0
      SUM1=0.0
      DO 10 I=1,N
  10  SUM=SUM+X(I)
      M0=SUM/FLOAT(N)
C*************************************************************************
C
      M=N-1
      DO 20 I=1,M
      C=FLOAT(N-I)*X(I)
      SUM1=SUM1+C
  20  CONTINUE
C
C*************************************************************************
```

```
      M1=SUM1/FLOAT(N*(N-1))
      A(IS)=ALOG(2.0)/(M0-2*M1)
      B(IS)=M0-0.57721/A(IS)
      RETURN
      END
C
C-----------------------------------------------------------------------
      SUBROUTINE ET(IS)
      COMMON/PARA/X(1200),N
      COMMON/RESUL/A(7),B(7)
C-----------------------------------------------------------------------
C
C     THIS PROGRAM CALCULATES THE PARAMETERS OF EV1 DISTRIBUTION
C
C     BY USING METHOD OF MAXIMUM ENTROPY PRINCIPLE
C
C
C*************************************************************************
C
C                   PARAMETERS ARE A AND B
C
C*************************************************************************
C
C                    START OF MAIN PROGRAM
C
C*************************************************************************
      DIMENSION Z(1100),EZ(1100)
      REAL NU
      PI=3.14159
C*************************************************************************
C
      SUM=0.0
      SUM1=0.0
      DO 10 I=1,N
   10 SUM=SUM+X(I)
      XBAR=SUM/FLOAT(N)
      DO 20 I=1,N
      STA=(X(I)-XBAR)**2
      SUM1=SUM1+STA
   20 CONTINUE
      VAR=SUM1/FLOAT(N-1)
      SD=SQRT(VAR)
      ALPHA=SD*SQRT(6.0)/(PI)
      U=XBAR-(.5772*ALPHA)
   21 SUM2=0.0
      SUM3=0.0
      DO 30 I=1,N
      Z(I)=(X(I)-U)/ALPHA
   30 SUM2=SUM2+Z(I)
      ZBAR=SUM2/FLOAT(N)
      DO 40 I=1,N
      EZ(I)=EXP(-Z(I))
      SUM3=SUM3+EZ(I)
   40 CONTINUE
      EZBAR=SUM3/FLOAT(N)
      BETA=ALOG(EZBAR)+ZBAR+.4228
      NU=ZBAR-.5772*BETA
      IF ((ABS(1.-BETA) .LE. 1.E-5) .AND. (ABS(NU) .LE. 1.E-5))GO TO 41
      ALPHA=ALPHA*BETA
      U=U+ALPHA*NU
```

```
      GO TO 21
 41   A(IS)=1./ALPHA
      B(IS)=U
      RETURN
      END
C
C-----------------------------------------------------------------------
C
C   SUBROUTINE LEAST CALCULATES THE PARAMETERS OF GUMBEL
C
C   DISTRIBUTION BY THE PRINCIPLE OF LEAST SQUARES
C
C-----------------------------------------------------------------------
C
C
      SUBROUTINE LEAST(IS)
      COMMON/PARA/X(1200),N
      COMMON/RESUL/A(7),B(7)
      SUM=0.0
      SUM1=0.0
      SUM2=0.0
      SUM3=0.0
      DO 100 I=1,N
      P= (FLOAT(I)-0.44)/(FLOAT(N)+0.12)
      Z=ALOG(-ALOG(P))
      Y=X(I)*Z
      SUM=SUM+Y
      SUM1=SUM1+Z
      SUM2=SUM2+X(I)
      SUM3=SUM3+X(I)**2
 100  CONTINUE
      A(IS)=((FLOAT(N)*SUM)-(SUM2*SUM1))/((SUM2**2)-(FLOAT(N)*SUM3))
      B(IS)=(SUM1+A(IS)*SUM2)/(A(IS)*FLOAT(N))
      RETURN
      END
C
C-----------------------------------------------------------------------
      SUBROUTINE MMM(IS)
C-----------------------------------------------------------------------
C
C     THIS PROGRAM CALCULATES THE PARAMETERS OF EV1 DISTRIBUTION
C     BY USING METHOD OF MIXED MOMENTS
C
C                  PARAMETERS ARE A AND B
C
C-----------------------------------------------------------------------
      COMMON/PARA/X(1200),N
      COMMON/RESUL/A(7),B(7)
      REAL M0,M1
C
      SUM=0.0
      SUM1=0.0
      SUM2=0.0
      DO 10 I=1,N
  10  SUM=SUM+X(I)
      M0=SUM/FLOAT(N)
C
      DO 20 I=1,N
      C=(X(I)-M0)*(X(I)-M0)
      SUM1=SUM1+C
```

```
      SUM2=SUM2+X(I)*X(I)
   20 CONTINUE
      VAR=SUM1/FLOAT(N-1)
      SX=SQRT(VAR)
      M1=SUM2/FLOAT(N)
C
C*************************************************************************
      A(IS)=1.2825498/SX
      B(IS)=(ALOG(1.0+(A(IS)*M0)+(A(IS)*A(IS)*M1/2.0)))/A(IS)
      RETURN
      END
C
C-------------------------------------------------------------------------
C
C      SUBROUTINE TO CALCULATE PARAMETERS OF GUMBEL
C
C      DISTRIBUTION BY METHOD OF INCOMPLETE MEANS
C
C-------------------------------------------------------------------------
      SUBROUTINE IM(IS,ICOUNT)
      COMMON/PARA/X(1200),N
      COMMON/RESUL/A(7),B(7)
      DIMENSION XM(100),NN(4)
C     WRITE(6,*)(X(I),I=1,N)
           ICOUNT=0
      L=1
      XMEAN=X(1)-1.0
      DO 15 IDUM=1,3
      DO 16 M=L,N
      IF (X(M).LT.XMEAN) GO TO 16
      SUM1=0.
      L=M
C             WRITE(6,*)N,A(1)
      DO 12 K=M,N
   12 SUM1=SUM1+X(K)
      GO TO 17
   16 CONTINUE
   17 XMEAN=SUM1/FLOAT(N-L+1)
      XM(IDUM)=XMEAN
C             WRITE(6,*)XM(IDUM)
      IF(IDUM.EQ.1) GO TO 15
      NN(IDUM)=L-1
   15 CONTINUE
      XBAR=XM(1)
      XBAR1=XM(2)
      XBAR2=XM(3)
      N1=NN(2)
      IF(N1.EQ.0)N1=1
      N2=NN(3)
      IF(N2.EQ.0)N2=1
      IF(XBAR1.EQ.XBAR2)GO TO 75
      GO TO 76
   75 ICOUNT=1
      A(IS)=0.0
      B(IS)=0.0
      GO TO 77
C     WRITE(6,*)XBAR,ICOUNT
C     WRITE(6,*)XBAR1,N1
C     WRITE(6,*)XBAR2,N2
   76 V=ALOG(FLOAT(N)/FLOAT(N1))
```

```
C             WRITE(6,*)N,B(1),V
      U=ALOG(FLOAT(N)/FLOAT(N2))
C             WRITE(6,*)N,B(1),U
      Q=((V*ALOG(V)/24.)*(24.-12*V+4.*V**2-V**3))-(V/288.)*
     1(288.-72.*V+16.*V**2-3*V**3)
C             WRITE(6,*)N,B(1)
      P=((U*ALOG(U)/24.)*(24.-12*U+4.*U**2-U**3))-(U/288.)*
     1(288.-72.*U+16.*U**2-3*U**3)
C             WRITE(6,*)N,B(1),P,XBAR1,XBAR2,N2,N1
      A(IS)=FLOAT(N)/(XBAR1-XBAR2)*(P/FLOAT(N-N2)-Q/FLOAT(N-N1))
C             WRITE(6,*)N,B(1),A(IS)
      B(IS)=XBAR1+(FLOAT(N)*Q)/(A(IS)*(FLOAT(N-N1)))
C77   WRITE(6,*)ICOUNT,A(IS),B(IS)
 77   RETURN
      END
C
C
C
C*************************************************************************
C
C  THIS PROGRAM CALCULATES THE K S STATISTIC FOR EV1 DISTRIBUTION
C
C                     SUBROUTINE   KSTEST
C
C*************************************************************************
C
      SUBROUTINE KSTEST
      COMMON/PARA/X(1200),N
      COMMON/RESUL/A(7),B(7)
      COMMON/TEST/C(7),CRIT
      DIMENSION G(100)
      CRIT=0.21
      IF (N.GT.50)GO TO 40
  40  CRIT=1.36/SQRT(FLOAT(N))
      DO  10 I=1,7
      H=0.0
      DO 20 J=1,N
      F=(FLOAT(J)-0.44)/(FLOAT(N)+0.12)
      Y=A(I)*(X(J)-B(I))
      P=EXP(-(EXP(-Y)))
      G(I)=ABS(F-P)
      IF (G(I).GE.H) H=G(I)
 20   CONTINUE
      C(I)=H
 10   CONTINUE
      END
C
C
      SUBROUTINE SURP
C
C-------------------------------------------------------------------
C
C  THIS PROGRAM CALCULATES THE MAXIMUM ENTROPY FOR TESTING
C
C  METHOD OF PARAMETER ESTIMATION
C
C-------------------------------------------------------------------
C
      COMMON/PARA/X(1200),N
      COMMON/RESUL/A(7),B(7)
```

```
      COMMON/STAT/XAVG,SIGMA,CS,TS
      COMMON/ENTR/ENT(7)
      PI=3.1415927
      C=(SQRT(2*PI))*SIGMA
      D=ALOG(1.0/C)
      E=(1.0/(2.0*SIGMA**2))*D*1.2825**2
      WRITE (6,*)C,D,E
      DO 10 J=1,7
      ENT(J)=A(J)*XAVG-A(J)*B(J)+1.0-ALOG(A(J))+(E/A(J)**2)
   10 CONTINUE
      RETURN
      END
C
      SUBROUTINE CI(ID)
C-----------------------------------------------------------------------
C
C  THIS PROGRAM CALCULATES THE CLASS INTERVALS FOR GUMBEL
C
C   DISTRIBUTION ALSO FINDS THE NUMBER OF FLOOD VALUES
C
C       FOR THAT INTERVAL FOR CHI SQUARE TEST
C
C-----------------------------------------------------------------------
C
      COMMON/PARA/X(1200),N
      COMMON/RESUL/A(7),B(7)
      COMMON/CLASS/F(42)
      DIMENSION P(10),E(10),ID(10,10)
      K=1
      P(1)=.14286
      P(2)=.28571
      P(3)=.42857
      P(4)=.57143
      P(5)=.71429
      P(6)=.85714
C     READ(5,*)(P(I),I=1,6)
      DO 15 I=1,7
      IXP=0
      JJ=1
      DO 14 J=1,6
      E(J)=B(I)-(ALOG(-ALOG(P(J)))/A(I))
      F(K)=E(J)
      DO 12 L=JJ,N
      IF (X(L).LE.E(J))GO TO 12
      ID(I,J)=L-1-IXP
      IXP=L-1
      GO TO 13
   12 CONTINUE
   13 JJ=L-1
      K=K+1
   14 CONTINUE
      ID(I,7) = N-L+1
   15 CONTINUE
      RETURN
      END
C
C
      SUBROUTINE CHI(IO)
C
C-----------------------------------------------------------------------
```

```
C
C  THIS PROGRAM CALCULATES THE CHI SQUARE STATISTIC FOR GUMBEL
C
C                          DISTRIBUTION
C
C-----------------------------------------------------------------------
C
C
      COMMON/PARA/X(1200),N
      COMMON/CHIS/R(7)
      DIMENSION IO(10,10)
      DO 50 K=1,7
      E=FLOAT(N)/7.0
      SUM=0.0
      DO 10 J=1,7
      DEV=(FLOAT(IO(K,J))-E)**2
 10   SUM=SUM+DEV
      SUM1=SUM/E
      R(K)=SUM1
 50   CONTINUE
      RETURN
      END
C
C
C-----------------------------------------------------------------------
C
C               PROGRAM TO GENERATE CONFIDENCE INTERVAL
C
C               FOR EXPONENTIAL DISTRIBUTION (EV1)
C
C-----------------------------------------------------------------------
C
C                             MAIN PROGRAM
C
C-----------------------------------------------------------------------
      SUBROUTINE CONINT
      COMMON/PARA/X(1200),N
      COMMON/STAT/XAVG,SIGMA,CS,TS
      COMMON/INTER/CIUP(6),CILO(6),T(6)
      DIMENSION VARX(10),XSTAR(10),G(10)
C-----------------------------------------------------------------------
C
C
C
C
C
C-----------------------------------------------------------------------
C
      T(1)=2.
      T(2)=5.
      T(3)=10.
      T(4)=20.
      T(5)=50.
      T(6)=100.
      DO 10 J=1,6
      G(J)=-(.45+.7797*ALOG((ALOG(T(J))-ALOG((T(J)-1.)))))
      VARX(J)=(SIGMA**2/FLOAT(N))*(.6+.5*FLOAT(N)/FLOAT(N-1))*
     1(1.0+1.14*G(J)+G(J)**2)
      XSTAR(J)=XAVG+SIGMA*G(J)
      CIUP(J)=XSTAR(J)+1.96*SQRT(VARX(J))
```

```
      CILO(J)=XSTAR(J)-1.96*SQRT(VARX(J))
   10 CONTINUE
      RETURN
      END
C
C=======================================================================
C
C THIS SUBROUTINE CALCULATES THE BIAS FOR GUMBEL DISTRIBUTION
C
C=======================================================================
C
      SUBROUTINE BIASA
      DIMENSION U(10)
      COMMON/PARA/X(1200),N
      COMMON/RESUL/A(7),B(7)
      COMMON/PEST/BIAS(7)
      T=(FLOAT(N)+0.12)/(FLOAT(1)-0.44)
      P=1.-(1./T)
      Y=-(ALOG(-ALOG(P)))
      DO 10 I=1,7
      U(I)=(Y/A(I))+B(I)
      BIAS(I)=(U(I)-X(N))/X(N)
   10 CONTINUE
      RETURN
      END
C
C
C=======================================================================
C
C  SUBROUTINE TO CALCULATE MEAN SQUARE ERROR FOR
C
C  GUMBEL DISTRIBUTION
C
C=======================================================================
      SUBROUTINE RMSE
      DIMENSION V(100),R(100),S(100)
      COMMON/PARA/X(1200),N
      COMMON/RESUL/A(7),B(7)
      COMMON/ERR/ERROR(7),ERRO(7)
      DO 10 J=1,7
      SUM=0.0
      SUM1=0.0
      K=N
      DO 20 I=1,N
      T=(FLOAT(N)+0.12)/(FLOAT(K)-0.44)
      P=1.-(1./T)
      Y=-(ALOG(-ALOG(P)))
      V(I)=(Y/A(J))+B(J)
      R(I)=((V(I)-X(I))/X(I))**2
      S(I)=(ABS(V(I)-X(I)))/X(I)
      SUM1=SUM1+S(I)
      SUM=SUM+R(I)
      K=K-1
   20 CONTINUE
      ERROR(J)=(SUM/FLOAT(N))*100.0
      ERRO(J)=(SUM1/FLOAT(N))*100.0
   10 CONTINUE
      RETURN
      END
C-----------------------------------------------
```

```
      REAL FUNCTION FF(ALPHA)
      COMMON/PARA/X(1200),N
      COMMON/STAT/XAVG,SIGMA,CS,TS
      S1=0.0
      S2=0.0
      DO 10 I=1,N
      TERM=EXP(-ALPHA*X(I))
      S1=S1+TERM
      S2=S2+TERM*X(I)
 10   CONTINUE
      FF=XAVG-(S2/S1)-(1.0/ALPHA)
      RETURN
      END
//*//GO.FT08F001 DD DSN=CEAROR.OUT.G21,DISP=SHR
//*//GO.FT09F001 DD DSN=CEAROR.OUT.G22,DISP=SHR
//*//GO.FT10F001 DD DSN=CEAROR.OUT.G23,DISP=SHR
//*//GO.FT11F001 DD DSN=CEAROR.OUT.G24,DISP=SHR
//GO.SYSIN DD *
MOMMLEPWMENTLEAMIXICM
.001 .01 .02 .05 .10 .25 .50 .75 .90 .95 .98 .99 .999
100
//
```

```
//PROJE9 JOB (1304,59634,11,20),'ARORA',MSGCLASS=S,CLASS=D,                00010000
//     NOTIFY=CEAROR                                                       00011000
/*ROUTE PRINT CEBA                                                        00020000
//     EXEC WATFIV,REGION.GO=4000K,TIME.GO=200                            00030000
/*JOBPARM SHIFT=N                                                         00031000
$JOB           TIME=660                                                  00040000
C                                                                        00050000
C*************************************************************************00060000
C                                                                        00061000
C     THIS PROGRAM CALCULATES THE BIAS OF ESTIMATORS 1/A AND B OF        00070000
C     GUMBEL DISTRIBUTION                                                00080000
C                                                                        00090000
C*************************************************************************00100000
C                                                                        00110000
      DOUBLE PRECISION DSEED1,DSEED                                      00120000
      INTEGER SSIZE(15)                                                  00130000
      DIMENSION RA(500)                                                  00131000
      REAL MA,MB,MA2,MB2,MSR,MSR1                                        00140000
      COMMON/DATA/X(500),NR                                              00180000
      COMMON/PARAM/A,B                                                   00190000
C                                                                        00200000
C----------------      INITIALISATION      ----------------------------00201000
C                                                                        00202000
            APP=0.01                                                     00220000
            BPP=200.0                                                    00230000
C                                                                        00230300
            DSEED1=729175.D0                                             00231000
C                                                                        00232100
            NCASE=12                                                     00233000
            NS1=25000                                                    00233200
            NS2=10000                                                    00233300
C                                                                        00233400
C----------------      WRITE INPUT PARAMETERS-------------------------00234000
C                                                                        00235000
      WRITE(6,543)APP,BPP                                                00240000
      WRITE(6,191)DSEED1                                                 00241000
C                                                                        00242000
 543  FORMAT(/,5X,'GUMBEL PARAMETERS : ',2X,'A = ',F6.3,/,               00250000
     1'                          B = ',F7.3,/)                           00260000
 191  FORMAT(/,2X,'INITIAL SEED = ',D23.16,/)                            00340000
C                                                                        00350000
C----------------      READ SAMPLE SIZES     -------------------------00360000
C                                                                        00370000
      READ(5,*)(SSIZE(I),I=1,NCASE)                                      00380000
      WRITE(6,*)(SSIZE(I),I=1,NCASE)                                     00380100
C                                                                        00381000
C---------     OUTER LOOP '51' FOR VARIOUS SAMPLE SIZES    ------------00390000
C                                                                        00391000
      DO 51 IS=1,NCASE                                                   00400000
C                                                                        00401000
           NR=SSIZE(IS)                                                  00410000
           DSEED=DSEED1                                                  00420000
C                                                                        00430000
           IF(NR.GT.100)GO TO 789                                        00440000
```

```
          NS=NS1                                                        00450000
          GO TO 788                                                     00460000
 789      NS=NS2                                                        00470000
C                                                                       00471000
 788      WRITE(6,155)NS                                                00480000
 155      FORMAT(130('*'),//,2X,'NO. OF SAMPLES = ',I5,/)               00490000
C                                                                       00491000
              SA=0.0                                                    00510000
              SB=0.0                                                    00520000
              SA2=0.0                                                   00530000
              SB2=0.0                                                   00540000
              SR=0.0                                                    00560000
C- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -00561000
C-LOOP '50' FOR GENERATING 'NS' NO. OF SAMPLES OF SAMPLE SIZE 'NR' EACH-00570000
C                                                                       00571000
          DO 50 I=1,NS                                                  00580000
               CALL GGUBS(DSEED,NR,RA)                                  00590000
                    DO 111 J =1,NR                                      00610000
                    RAA=RA(J)                                           00620000
  111               X(J)=BPP-(ALOG(-ALOG(RAA)))/APP                     00630000
C                                                                       00640000
               CALL VSRTA(X,NR)                                         00650000
C                                                                       00680000
               CALL MM                                                  00690000
C                                                                       00722000
                   SA=SA+A                                              00730000
                   SB=SB+B                                              00740000
                   SA2=SA2+A*A                                          00750000
                   SB2=SB2+B*B                                          00760000
                   SR=SR+APP/A                                          00770000
 50       CONTINUE                                                      00780000
C                        .... LOOP '50' ENDS ....                       00781000
C- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -00790000
C                                                                       00800000
C-------------    CALCULATION OF ESTIMATOR STATISTIC   ----------------00801000
C                                                                       00802000
              XNS=FLOAT(NS)                                             00803000
C ... MEAN :                                                            00804000
              MA=SA/XNS                                                 00810000
              MB=SB/XNS                                                 00820000
              MA2=SA2/XNS                                               00830000
              MB2=SB2/XNS                                               00840000
              MSR=SR/XNS                                                00850000
              MSR1=1.0-MSR                                              00860000
C ... STD. DEVIATION :                                                  00861000
              STDA=(MA2-MA*MA)**0.5                                     00870000
              STDB=(MB2-MB*MB)**0.5                                     00880000
C ... BIAS :                                                            00881000
              BA=APP-MA                                                 00890000
              BB=BPP-MB                                                 00900000
C ... RELATIVE BIAS :                                                   00901000
              SBA=BA/APP                                                00910000
C                                                                       00920000
C---------------------     WRITE RESULTS      -------------------------00921000
C                                                                       00922000
          WRITE(6,*)'    E(1-A/ACUP) = ',MSR1                           00930000
          WRITE(6,98)                                                   00940000
 98       FORMAT(//,1X,'METHOD  SAMPLE SIZE   BIAS(A)       STD(A)     00950000
     *      BIAS(B)      STD(B)     ',/,1X,80('-'),/)                  00960000
          WRITE(6,99)NR,BA,STDA,BB,STDB                                 00970000
```

```
 99        FORMAT(10X,I4,5X,4(F11.6,4X))                                  00980000
 51    CONTINUE                                                           01000000
       STOP                                                               01010000
       END                                                                01020000
C                                                                         01030000
C///////////////    END OF MAIN PROGRAM     ////////////////////////////01031000
C////////////////////////////////////////////////////////////////////////01040000
C                                                                         01430000
C------------------      SUBROUTINE MM      ----------------------------01440000
C                                                                         01450000
       SUBROUTINE MM                                                      01560000
       COMMON/DATA/X(500),NR                                              01570000
       COMMON/PARAM/A,B                                                   01580000
       REAL M1,M2,M3,K                                                    01590000
C      WRITE(6,*)(X(I),I=1,N)                                             01610000
       XN=FLOAT(NR)                                                       01620000
       AP=0.0                                                             01630000
       BP=0.0                                                             01640000
C                                                                         01650000
       DO 1 I=1,NR                                                        01660000
       AP=AP+X(I)                                                         01670000
       BP=BP+X(I)*X(I)                                                    01680000
 1     CONTINUE                                                           01700000
C                                                                         01701000
       M1=AP/XN                                                           01710000
       M2=(BP/XN)-M1*M1                                                   01720000
       M2=M2*XN/(XN-1.0)                                                  01730000
C                                                                         01740000
       A=3.1415927/(SQRT(M2*6))                                           01760000
       B=M1-0.57721/A                                                     01780000
C                                                                         01790000
       RETURN                                                             01880000
       END                                                                01890000
$ENTRY                                                                    01910000
5 7 10 15 20 30 40 50 75 100 150 200                                      01920000
$$                                                                        01921000
//                                                                        01930000
.001 .01 .02 .05 .10 .25 .50 .75 .90 .95 .98 .99 .999                     01940000
```

```
//PROJECT JOB (1304,77493,1,20),'ARORA',MSGCLASS=S,CLASS=B
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=D
//     EXEC WATFIV,TIME.GO=99
$JOB           TIME=360
C=====================================================================
C   THIS PROGRAM CHECKS THE BIAS IN STATISTIC XAVG =(X1+X2+ ... +XN)/N
C   FOR GUMBEL'S MONTE-CARLO SAMPLES AS N INCREASES.
C=====================================================================
      DOUBLE PRECISION DSEED,RR,A,B,X,DL,SUM,SR
      DIMENSION R(101000)
      DSEED=234567.D0
      WRITE(6,*)'     INITIAL SEED= ',DSEED
      WRITE(6,*)'                                   '
      SUM=0.0D0
      SR=0.0D0
C
      A=1.0D0
      B=0.0D0
C
      A1=1.0
      B1=0.0
      EX=B1+0.57721/A1
      WRITE(6,*)'            E(X) =',EX
      WRITE(6,*)'                                   '
      WRITE(6,*)'                                   '
      N=25000
      DO 10 I=1,16
      CALL GGUBS(DSEED,N,R)
            DO 20 J=1,N
            RR=R(J)
            SR=SR+RR
            DL=DLOG(-DLOG(RR))
            X=B-DL/A
            SUM=SUM+X
 20         CONTINUE
      TNV=FLOAT(N)*FLOAT(I)
      NV=N*I
      RAVG=SR/TNV
      XAVG=SUM/TNV
      BIASX=EX-XAVG
      WRITE(6,*)'    N = ',NV,'        XAVG = ',XAVG,'        BIAS = ',B
     * IASX,'    AVG(F) = ',RAVG
      WRITE(6,*)'                                   '
 10   CONTINUE
      STOP
      END
$ENTRY
$$
//
```

```
//PROJECT JOB (1304,77493,5,20),'ARORA',MSGCLASS=S,CLASS=B
/*ROUTE PRINT CEBA
//     EXEC FORTVCLG,REGION.GO=4000K,TIME.GO=200
/*JOBPARM SHIFT=D
//FORT.SYSIN DD *
C
C  ************************************************************************
C  *           TWO COMPONENT EXTREME VALUE DISTRIBUTION                   *
C  * ---------------------------------------------------------------------*
C  *     THIS PACKAGE COMPUTES STATISTICAL PROPERTIES OF ENTROPY EST-      *
C  *  IMATORS OF PARAMETERS AB1, AND ADL1 OF TWO COMPONENT EXTREME         *
C  *  VALUE DISTRIBUTION. FIRST OF ALL A LARGE NUMBER OF TCEV SAMPLES      *
C  *  ARE GENERATED FROM A GIVEN POPULATION. THEN THE ENTROPY ESTIMATE     *
C  *  OF THE PARAMETERS ARE COMPUTED. THESE ESTIMATES ARE USED TO          *
C  *  PREDICT THE STATISTICAL PROPERTIES OF THE ESTIMATORS.                *
C  *.....................................................................*
C  *      THE PACKAGE USES THE FOLLOWING SUBROUTINES :                    *
C  *          1) ROOT -- CALCULATES THE ROOTS/ENTROPY ESTIMATES OF        *
C  *                     A SAMPLE ; USES IMSL ROUTINE "ZBRENT" FOR        *
C  *                     CALCULATION OF ROOT OF F(AB1)=0*                 *
C  *          2) TCEVVA -- CALCULATES QUANTILE ESTIMATE OF A SAMPLE       *
C  *      FUNCTION SUBPROGRAMS USED ARE :                                 *
C  *          1A) FF(THETA) -- USED BY IMSL ROUTINE "ZBRENT"              *
C  *          1B) ESUM(XX) -- USED IN SUBROUTINE ROOT TO COMPUTE          *
C  *                          THE SAMPLE ESTIMATE OF ADL1                 *
C  *           2) FX(X), F1X(X) -- USED IN SUBROUTINE TCEVVA              *
C  *.....................................................................*
C  *      IMPORTANT VARIABLES LIST :                                      *
C  *          1) NSAMPL    = NO. OF TCEV SAMPLES GENERATED                *
C  *          2) AB1,ADL1  = PARAMETERS THETA1 AND LAMDA1 RESPECTIVLY     *
C  *          3) X(.)      = TCEV SAMPLE VALUES                           *
C  *          4) XAVG      = TCEV SAMPLE AVERAGE                          *
C  *          5) FPR(4)    = CUM. PROBABILITIES FOR QUANTILES QNTL(4)     *
C  *          6) QNTL(4)   = QUANTILES OF AB1 & ADL1 (AT CUM.             *
C  *                                          PROBABILITIES FPR(4))       *
C  *          7) PPQNTL(4) = POPULATION QUANTILE VALUES (  - DO -  )      *
C  *          8) 'S...' IN FRONT OF A VARIABLE IMPLIES SUMMATION, E.G.    *
C  *                     SQNTL(.) = SUM OF ALL QUANTILE ESTIMATES         *
C  *                     FROM 'NSAMPL' NO. OF SAMPLES                     *
C  *          9) 'M...' IN FRONT OF A VARIABLE IMPLIES MEAN OF THAT       *
C  *                     STATISTIC, E.G. MAB1 = MEAN OF AB1,E.T.C.        *
C  *         10) 'S2...' --  SQUARE SUM OF THE VARIABLE                   *
C  *         11) 'MS...' --  MEAN SQUARE ERROR OF THE VARIABLE            *
C  *         12) 'MN...' --  MINIMUM VALUE OF THE VARIABLE.               *
C  *         13) 'MX...' --  MAXIMUM VALUE OF THE VARIABLE.               *
C  *         14) 'B....' --  BIAS OF THE STATISTIC                        *
C  *         15) 'V....' --  VARIANCE OF THE VARIABLE                     *
C  ************************************************************************
C
      DOUBLE PRECISION DSEED1,DSEED2,DS1,DS2
      DIMENSION X1(50),X2(50),FPR(4),QNTL(4),SQNTL(4)
      DIMENSION PPQNTL(4),S2QNTL(4),BQNTL(4),VQNTL(4)
      COMMON/PARA/X(50),C
      COMMON/STAT/XAVG
```

```
      COMMON/CONST/A1
      REAL MAB1,MADL1,MQNTL(4),MSAB1,MSADL1,MSQNTL(4)
      REAL MXAB1,MNAB1,MXADL1,MNADL1
      DATA FPR(1),FPR(2),FPR(3),FPR(4)/0.5,0.9,0.99,0.999/
      DATA PPQNTL(1),PPQNTL(2),PPQNTL(3),PPQNTL(4)/28.9965515,54.4592285
     *,110.845917,181.138016/
C
C----------------------   INITIALIZATION    ---------------------------
C
      NSAMPL=5000
      DSEED1=654321.D0
      DS1=DSEED1
      DSEED2=987001.D0
      DS2=DSEED2
C
      SAB1=0.0
      SADL1=0.0
      S2AB1=0.0
      S2ADL1=0.0
      DO 7 I=1,4
      QNTL(I)=0.0
      SQNTL(I)=0.0
      S2QNTL(I)=0.0
  7   CONTINUE
C
      AB1P=10.0
      ADL1P=10.0
      AB1PP=10.0
      ADL1PP=10.0
C
C-----   CALCULATION OF CONSTANTS 'C' AND 'A1' IN THE FUNCTION ------
C
      A=0.0
      B=0.0
      F=1.0
      SIGN=-1.0
      J=1
  10  XJ=FLOAT(J)
      ARG=XJ/3.067
      G=GAMMA(ARG)
      F=F*(0.1734/XJ)*SIGN
      FF=F*G
      FF1=FF*XJ
      IF(ABS(FF1).LE.1.0E-08)GO TO 20
      A=A+FF
      B=B+FF1
      J=J+1
      GO TO 10
C.............CONSTT A1.............
  20  A1=40.0*(1.+B/3.067)
      ALG=ALOG(A1)
C.............CONSTT C..............
      C=ALG+0.5772-A
C     WRITE(6,*)C,F1,F,J
C
C* * * * * START OF OUTER LOOP 101 FOR NO. OF SAMPLES  * * * * *
C
      DO 101 NS=1,NSAMPL
C
C
```

```
C-----------------     SAMPLE GENERATION      -------------------------
C
C
      CALL GGUBS(DSEED1,40,X1)
      CALL GGUBS(DSEED2,40,X2)
      DO 1 I=1,40
      X1(I)=ALOG(-ALOG(X1(I))/10.0)*(-10.0)
      X2(I)=ALOG(-ALOG(X2(I))/0.3673674)*(-30.67)
      IF(X1(I).GE.X2(I))GO TO 2
      X(I)=X2(I)
      GO TO 1
  2   X(I)=X1(I)
C 3   WRITE(6,*)X1(I),X2(I),X(I)
  1   CONTINUE
C     WRITE(6,*)(X(I),I=1,40)
C
C------------------    SAMPLE STATISTICS      -------------------------
C
      XAVG=0.0
      DO 5 I=1,40
  5   XAVG=XAVG+X(I)
      XAVG=XAVG/40.0
C
C-------   ENTROPY PARAMETER ESTIMATION
C          AND THEIR SUMMATIONS     ------------------------------------
C
      AB11=2.0
      AB1=40.0
      CALL ROOT(AB1,AB11,ADL1)
C         WRITE(6,*)AB1,ADL1
      SAB1=SAB1+AB1
      SADL1=SADL1+ADL1
      S2AB1=S2AB1+AB1*AB1
      S2ADL1=S2ADL1+ADL1*ADL1
C
C-------   CALCULATION OF QUANTILES FROM
C          ESTIMATED PARAMETERS      ----------------------------------
C
      AB2=3.067*AB1
      ADL2=0.1734*(ADL1**(1.0/3.067))
      DO 12 I=1,4
      FF=FPR(I)
      CALL TCEVVA(ADL1,ADL2,AB1,AB2,FF,XX)
      QNTL(I)=XX
      SQNTL(I)=SQNTL(I)+QNTL(I)
      S2QNTL(I)=S2QNTL(I)+QNTL(I)*QNTL(I)
  12  CONTINUE
C         WRITE(6,*)(QNTL(J),J=1,4)
C
C--------------    CALCULATION OF EXTREME ESTIMATORS    ---------------
C
      MXAB1=AMAX1(AB1,AB1P)
      MNAB1=AMIN1(AB1,AB1PP)
      AB1P=MXAB1
      AB1PP=MNAB1
      MXADL1=AMAX1(ADL1,ADL1P)
      MNADL1=AMIN1(ADL1,ADL1PP)
      ADL1P=MXADL1
      ADL1PP=MNADL1
C
```

```
C
 101  CONTINUE
C
C* * * * * * * * *        END OF LOOP 101            * * * * * * * * * *
C
C------------------     ESTIMATOR STATISTICS     ----------------------
C
      MAB1=SAB1/NSAMPL
      MADL1=SADL1/NSAMPL
      VAB1=(S2AB1/NSAMPL)-(MAB1*MAB1)
      VADL1=(S2ADL1/NSAMPL)-(MADL1*MADL1)
      BAB1=MAB1-10.0
      BADL1=MADL1-10.0
      MSAB1=BAB1*BAB1+VAB1
      MSADL1=BADL1*BADL1+VADL1
C
      DO 13 I=1,4
      MQNTL(I)=SQNTL(I)/NSAMPL
      VQNTL(I)=(S2QNTL(I)/NSAMPL)-(MQNTL(I)*MQNTL(I))
      BQNTL(I)=MQNTL(I)-PPQNTL(I)
      MSQNTL(I)=BQNTL(I)*BQNTL(I)+VQNTL(I)
  13  CONTINUE
C
C-------------------      PRINT RESULTS      --------------------------
C
          WRITE(6,198)DS1,DS2
 198  FORMAT(5X,'DSEED1 = ',D20.10,/,5X,'DSEED2 = ',D20.10,/)
          WRITE(6,199)NSAMPL
 199  FORMAT(7X,'NO. OF SAMPLES = ',I5,//)
          WRITE(6,200)
 200  FORMAT(10X,'PARAMETERS : ',/,10X,12('-'),//,
     *17X,'MEAN',5X,'BIAS',5X,'VARIANCE',5X,'MSE',7X,
     *'MIN.',5X,'MAX.',/,
     *80('-'))
          WRITE(6,201)MAB1,BAB1,VAB1,MSAB1,MNAB1,MXAB1
 201  FORMAT(/,4X,'AB1',8X,6(F7.3,3X),/)
          WRITE(6,202)MADL1,BADL1,VADL1,MSADL1,MNADL1,MXADL1
 202  FORMAT(4X,'ADL1',7X,6(F7.3,3X),//,80('-'),//)
          WRITE(6,203)
 203  FORMAT(/,10X,'QUANTILES : ',/,10X,12('-'),//,
     *5X,'F = ',20X,'0.5',10X,'0.9',9X,'0.99',8X,'0.999',/)
          WRITE(6,204)(PPQNTL(I),I=1,4)
 204  FORMAT(5X,'X(POPULATION)',7X,4(F8.3,5X),/,80('-'),/)
          WRITE(6,205)(MQNTL(I),I=1,4)
 205  FORMAT(5X,'MEAN          ',7X,4(F8.3,5X),/)
          WRITE(6,206)(BQNTL(I),I=1,4)
 206  FORMAT(5X,'BIAS          ',7X,4(F8.3,5X),/)
          WRITE(6,207)(VQNTL(I),I=1,4)
 207  FORMAT(5X,'VARIANCE      ',7X,4(F8.3,5X),/)
          WRITE(6,208)(MSQNTL(I),I=1,4)
 208  FORMAT(5X,'MSE           ',7X,4(F8.3,5X),/,80('-'))
C
      STOP
      END
C
C::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C::::::::::::           MAIN SEGMENT ENDS...        ::::::::::::::::::::
C::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C
C
```

```
C
      SUBROUTINE ROOT(AB1,AB11,ADL1)
      COMMON/CONST/A1
      EXTERNAL FF
C
C--------   ITERATIVE ESTIMATION OF AB1
C           USING IMSL ROUTINE ZBRENT  ------------
C
      EPS=1.0E-5
      NSIG=5
      MAXFN=100
      CALL ZBRENT(FF,EPS,NSIG,AB11,AB1,MAXFN,IER)
      ADL1=A1/ESUM(AB1)
C         WRITE(6,*)AB1,ADL1
C
      RETURN
      END
C
      REAL FUNCTION FF(THETA)
      COMMON/PARA/X(50),C
      COMMON/STAT/XAVG
      S=0.0
      DO 10 I=1,40
      TERM=EXP(-X(I)/THETA)
      S=S+TERM
 10   CONTINUE
      S=ALOG(S)
      FF=XAVG+THETA*(S-C)
C     WRITE(6,*)THETA,FF
      RETURN
      END
C
      REAL FUNCTION ESUM(XX)
      COMMON/PARA/X(50),C
      S=0.0
      DO 10 I=1,40
      TERM=EXP(-X(I)/XX)
      S=S+TERM
 10   CONTINUE
      ESUM=S
      RETURN
      END
C
C
      SUBROUTINE TCEVVA(ADL1,ADL2,AB1,AB2,FF,XX)
C=============================================================
C  SUBROUTINE PER IL CALCOLO DI X UNA VOLTA ASSEGNATO
C  UN VALORE DELLA F(X) DEL MODELLO TCEV
C=============================================================
      REAL*8 DL1,DL2,B1,B2,F,X,FX
      COMMON DL1,DL2,B1,B2
      DL1=ADL1
      DL2=ADL2
      B1=AB1
      B2=AB2
      F=FF
      X=B1*DLOG(DL1)-B1*ALOG(ALOG(1/.35))
10    FR=FX(X)
      IF(ABS(FR-F).LT.0.0000001) THEN
      GOTO 57
```

```
      ELSE
      X=X-(FR-F)/F1X(X)
      END IF
      GOTO 10
 57   CONTINUE
      XX=X
      RETURN
      END
C
      FUNCTION FX(X)
      REAL*8 FX
      REAL*8 DL1,DL2,B1,B2,X
      COMMON DL1,DL2,B1,B2
      FX=DEXP(-DL1*DEXP(-X/B1)-DL2*DEXP(-X/B2))
C     WRITE(6,*)FX
      RETURN
      END
C
      REAL FUNCTION F1X(X)
      REAL*8 DL1,DL2,B1,B2,X,FX
      COMMON DL1,DL2,B1,B2
      F1X=FX(X)*((DL1/B1)*DEXP(-X/B1)+(DL2/B2)*DEXP(-X/B2))
      RETURN
      END
//GO.SYSIN DD *
//
//*GO.FT08F001 DD DSN=CEAROR.OUT.G21,DISP=SHR
//*GO.FT09F001 DD DSN=CEAROR.OUT.G22,DISP=SHR
C
C
C--------   ITERATIVE ESTIMATION OF AB1
C           USING IMSL ROUTINE ZREAL2  ------------
C
C     EPS=1.0E-5
C     EPS2=1.0E-5
C     ETA=1.0E-2
C     NSIG=4
C     ITMAX=100
C     N1=1
C     THETA1=10.0
C     CALL ZREAL2(FF,EPS,EPS2,ETA,NSIG,N1,THETA1,ITMAX,IER)
C     STHETA1=STHETA1+THETA1
C     WRITE(6,*)THETA1,ITMAX
C
          WRITE(6,201)NSAMPL,MAB1,MADL1
          WRITE(6,202)MNAB1,MXAB1
          WRITE(6,203)MNADL1,MXADL1
          WRITE(6,204)(MQNTL(J),J=1,4)
 201  FORMAT(5X,'NO. OF SAMPLES     = ',I5,/,
     *       5X,'MEAN OF AB1 (THETA1)  = ',F13.5,/,
     *       5X,'MEAN OF ADL1 (LEMDA1) = ',F13.5)
 202  FORMAT(//,5X,'MIN. AB1 = ',F13.5,/,
     *          5X,'MAX. AB1 = ',F13.5)
 203  FORMAT(//,5X,'MIN. ADL1 = ',F13.5,/,
     *          5X,'MAX. ADL1 = ',F13.5)
 204  FORMAT(//,25X,'MEAN OF QUANTILES : ',///,5X,4(F13.5,2X))
```

```
//PROJECT JOB (1304,59634,5,20),'ARORA',MSGCLASS=S,CLASS=A
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=N
//     EXEC WATFIV,REGION.GO=4000K,TIME.GO=99
$JOB            TIME=4500
C=====================================================================
C              TWO COMPONENT EXTREME VALUE DISTRIBUTION
C---------------------------------------------------------------------
C           CALCULATION OF REGIONAL PARAMETERS AND QUANTILES
C=====================================================================
      DOUBLE PRECISION DSEED1,DSEED2
      REAL X1(50),X2(50),FPR(4),QNTL(4),SQNTL(4),AB1(40),ADL1(40)
      REAL PPQNTL(4),S2QNTL(4),BQNTL(4),VQNTL(4)
      COMMON/PARA/X(50,50),C
      COMMON/YDATA/Y(1700)
      COMMON/XSTAT/XAVG(40)
      COMMON/CONST/A1
      COMMON/SAMPC/NSAMPL
C     REAL MAB1,MADL1,MQNTL(4),MSAB1,MSADL1,MSQNTL(4)
C     REAL MXAB1,MNAB1,MXADL1,MNADL1
C     DATA FPR(1),FPR(2),FPR(3),FPR(4)/0.5,0.9,0.99,0.999/
C     DATA PPQNTL(1),PPQNTL(2),PPQNTL(3),PPQNTL(4)/28.9965515,54.4592285
C    *,110.845917,181.138016/
C
C------------------      INITIALIZATION      ----------------------------
C
      DSEED1=123457.D0
      DSEED2=612392.D0
C
      AB1P=10.0
      ADL1P=10.0
      AB1PP=10.0
      ADL1PP=10.0
C
      NREPT=7
      NBAD=0
C
C -------------------      NO. OF REPETITIONS      ----------------------
C
      INR=1
C
  888 DO 31 I=1,40
      AB1(I)=10.0
  31  ADL1(I)=10.0
C
      ABP=0.0
      ADLP=0.0
      AB=3.067
      ADL=0.173
C
C----------------         SAMPLE GENERATION       ----------------------
C
      WRITE(6,*)'  DSEED1 = ',DSEED1,'  DSEED2 = ',DSEED2
      DO 32 I=1,40
```

```
C
      CALL GGUBS(DSEED1,40,X1)
      CALL GGUBS(DSEED2,40,X2)
      DO 1 J=1,40
      X1LN=-ALOG(X1(J))/10.0
      X1(J)=ALOG(X1LN)*(-10.0)
      X2LN=-ALOG(X2(J))/0.3665199
      X2(J)=ALOG(X2LN)*(-30.67)
C
C     X1LN=-ALOG(X1(J))/10.0
C     X1(J)=ALOG(X1LN)*(-10.0)
C     X2LN=-ALOG(X2(J))/(0.2*(10.0**(1./6.)))
C     X2(J)=ALOG(X2LN)*(-60.00)
C
      IF(X1(J).GE.X2(J))GO TO 2
      X(I,J)=X2(J)
      GO TO 1
  2   X(I,J)=X1(J)
  1   CONTINUE
C     WRITE(6,*)(X(I,J),J=1,40)
  32  CONTINUE
C
C---------------          SAMPLE STATISTICS         ----------------------
C
      DO 4 I=1,40
  4   XAVG(I)=0.0
      DO 5 I=1,40
      DO 5 J=1,40
  5   XAVG(I)=XAVG(I)+X(I,J)
      DO 6 I=1,40
  6   XAVG(I)=XAVG(I)/40.0
C
C -----------------    MLE ESTIMATE OF AB1 AND ADL1    -----------------
C
C     DO 7 I=1,40
C     CALL ESTMLE(A,B,I)
C     AB1(I)=A
C     ADL1(I)=B
C 7   CONTINUE
C
C---------     TRANSFORMATION OF ALL SAMPLES TO POOLED DATA     ---------
C
      NITR=0
C
 222  ICOUNT=1
      DO 33 I=1,40
      DO 33 J=1,40
      Y(ICOUNT)=(X(I,J)/AB1(I))-ALOG(ADL1(I))
      ICOUNT=ICOUNT+1
  33  CONTINUE
C
C-------        PARAMETERS OF POOLED SAMPLE (EQNS (36),(37))    --------
C
C     CALL YROOT(AB,ADL)
C     CALL YROOT1(AB,ADL)
C     CALL YROOT2(AB,ADL)
C     CALL YROOT3(AB,ADL)
C     CALL YROOT4(AB,ADL)
C     CALL YROOT6(AB,ADL)
C
```

```
      CALL YROOT5(AB,ADL)
C
      NITR=NITR+1
C
      IF((ABS(AB-ABP).LE.1.0E-03).AND.(ABS(ADL-ADLP).LE.1.0E-03))
     *GO TO 999
      ABP=AB
      ADLP=ADL
C
      IF(NITR.GE.20)THEN
      NBAD=NBAD+1
      GO TO 888
      END IF
C
C-----     CALCULATION OF CONSTANT C=C(AB,ADL) IN THE FUNCTION
C             F(AB1(I))=0, AND A1=A1(AB,ADL) IN G(ADL1(I)=0    --------
C
      SIGN=-1.0
      A=0.0
      B=0.0
      F=1.0
      J=1
  10  XJ=FLOAT(J)
      ARG=XJ/AB
      G=GAMMA(ARG)
      F=(ADL/XJ)*F*SIGN
      FF=F*G
      FF1=FF*XJ
      IF(ABS(FF1).LE.1.0E-08)GO TO 20
      A=A+FF
      B=B+FF1
      J=J+1
      GO TO 10
  20  A1=40.0*(1.+B/AB)
      AA=ALOG(A1)
      C=AA+0.5772-A
C     WRITE(6,*)C,F1,F,J
C
C-------------         ENTROPY PARAMETER ESTIMATION
C                         AND THEIR SUMMATIONS     ------------------
C
      DO 34 NSAMPL=1,40
      APAB11=2.0
      APAB1=40.0
      CALL ROOT(APAB1,APAB11,APADL1)
      AB1(NSAMPL)=APAB1
      ADL1(NSAMPL)=APADL1
C         WRITE(6,*)APAB1,APADL1
  34  CONTINUE
              GO TO 222
C
 999  CONTINUE
      WRITE(8,*)AB,ADL
      WRITE(9,*)(AB1(I),I=1,40)
      WRITE(9,*)(ADL1(I),I=1,40)
C
      IF(INR.EQ.NREPT)GO TO 1111
      INR=INR+1
      GO TO 888
C
```

```
C
 1111 WRITE(6,*)'  NO. OF BAD SAMPLES = ',NBAD
      STOP
      END
C=====================================================================
C                        END OF MAIN PROGRAM
C=====================================================================
C:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C=====================================================================
C
C------------------        SUBROUTINE ROOT      ----------------------
C
      SUBROUTINE ROOT(AB1,AB11,ADL1)
      COMMON/CONST/A1
      EXTERNAL FF
C
C---------------     ITERATIVE ESTIMATION OF AB1
C                     USING IMSL ROUTINE ZBRENT    -------------------
C
      EPS=1.0E-5
      NSIG=5
      MAXFN=100
      CALL ZBRENT(FF,EPS,NSIG,AB11,AB1,MAXFN,IER)
      ADL1=A1/ESUM(AB1)
C         WRITE(6,*)AB1,ADL1
C
      RETURN
      END
C
      REAL FUNCTION FF(THETA)
      COMMON/PARA/X(50,50),C
      COMMON/XSTAT/XAVG(40)
      COMMON/SAMPC/NSAMPL
      S=0.0
      DO 10 I=1,40
      TERM=EXP(-X(NSAMPL,I)/THETA)
      S=S+TERM
   10 CONTINUE
      S=ALOG(S)
      FF=XAVG(NSAMPL)+THETA*(S-C)
C     WRITE(6,*)THETA,FF
      RETURN
      END
C
      REAL FUNCTION ESUM(XX)
      COMMON/PARA/X(50,50),C
      COMMON/SAMPC/NSAMPL
      S=0.0
      DO 10 I=1,40
      TERM=EXP(-X(NSAMPL,I)/XX)
      S=S+TERM
   10 CONTINUE
      ESUM=S
      RETURN
      END
C
C
      SUBROUTINE TCEVVA(ADL1,ADL2,AB1,AB2,FF,XX)
C=====================================================
```

```
C  SUBROUTINE PER IL CALCOLO DI X UNA VOLTA ASSEGNATO
C  UN VALORE DELLA F(X) DEL MODELLO TCEV
C=============================================================
      REAL*8 DL1,DL2,B1,B2,F,X,FX
      COMMON DL1,DL2,B1,B2
      DL1=ADL1
      DL2=ADL2
      B1=AB1
      B2=AB2
      F=FF
      X=B1*DLOG(DL1)-B1*ALOG(ALOG(1/.35))
10    FR=FX(X)
      IF(ABS(FR-F).LT.0.0000001) THEN
      GOTO 57
      ELSE
      X=X-(FR-F)/F1X(X)
      END IF
      GOTO 10
 57   CONTINUE
      XX=X
      RETURN
      END
C
      FUNCTION FX(X)
      REAL*8 FX
      REAL*8 DL1,DL2,B1,B2,X
      COMMON DL1,DL2,B1,B2
      FX=DEXP(-DL1*DEXP(-X/B1)-DL2*DEXP(-X/B2))
C     WRITE(6,*)FX
      RETURN
      END
C
      REAL FUNCTION F1X(X)
      REAL*8 DL1,DL2,B1,B2,X,FX
      COMMON DL1,DL2,B1,B2
      F1X=FX(X)*((DL1/B1)*DEXP(-X/B1)+(DL2/B2)*DEXP(-X/B2))
      RETURN
      END
C
C-------------------        SUBROUTINE YROOT       ----------------------
C
      SUBROUTINE YROOT(AB,ADL)
      COMMON/YPARM/YBAR,EXYBAR
      REAL WK(42),A(2),PAR(2)
      EXTERNAL FCN
C     IN=1600
C     CALL YSTAT(IN)
C     PAR(1)=YBAR-0.5772
C     PAR(2)=EXYBAR-1.0
      PAR(1)=0.0
      PAR(2)=0.0
      N=2
      NSIG=5
      ITMAX=500
      A(1)=AB
      A(2)=ADL
      WRITE(6,*)A(1),A(2)
      CALL ZSCNT(FCN,NSIG,N,ITMAX,PAR,A,FNORM,WK,IER)
      WRITE(6,*)FNORM,A(1),A(2)
      AB=A(1)
```

```
      ADL=A(2)
      RETURN
      END
C
C----------------        SUBROUTINE YROOT1         --------------------
C
      SUBROUTINE YROOT1(AB,ADL)
      COMMON/YPARM/YBAR,EXYBAR
                WRITE(6,*)AB,ADL
                IC=1
      IN=1600
      CALL YSTAT(IN)
      PAR1=YBAR-0.5772
      PAR2=EXYBAR-1.0
      SIGN=-1.0
   30 J=2
      FAC=-1.0*ADL
      S1=0.0
      S2=0.0
   10 XJ=FLOAT(J)
      ARG=XJ/AB
      G=GAMMA(ARG)
      FAC=(ADL/XJ)*FAC*SIGN
      FF=FAC*G
      FF1=FF*XJ
      IF(ABS(FF1).LE.1.0E-08)GO TO 20
      S1=S1+FF
      S2=S2+FF1
      J=J+1
      GO TO 10
   20 GAMAFN=GAMMA(1.0/AB)
      ABNXT=(S2-ADL*GAMAFN)/PAR2
   50 J=2
      FAC=-1.0*ADL
      S1=0.0
      S2=0.0
   60 XJ=FLOAT(J)
      ARG=XJ/ABNXT
      G=GAMMA(ARG)
      FAC=(ADL/XJ)*FAC*SIGN
      FF=FAC*G
      FF1=FF*XJ
      IF(ABS(FF1).LE.1.0E-08)GO TO 70
      S1=S1+FF
      S2=S2+FF1
      J=J+1
      GO TO 60
   70 GAMAFN=GAMMA(1.0/ABNXT)
      ADLNXT=(PAR1+S1)/GAMAFN
                WRITE(6,*)IC,ABNXT,ADLNXT
      IF((ABS(ABNXT-AB).LE.1.0E-05).AND.(ABS(ADLNXT-ADL).LE.1.0E-05))
     *GO TO 40
      AB=ABNXT
      ADL=ADLNXT
      IF(IC.GE.50)GO TO 40
      IC=IC+1
      GO TO 30
   40 AB=ABNXT
      ADL=ADLNXT
      RETURN
```

```
      END
C
C-------------------        SUBROUTINE YROOT2        -------------------
C
      SUBROUTINE YROOT2(AB,ADL)
      COMMON/YPARM/YBAR,EXYBAR
      COMMON/YDATA/Y(1700)
                WRITE(6,*)AB,ADL
                IC=1
      IN=1600
C     CALL YSTAT(IN)
C     PAR1=YBAR-0.5772
C     PAR2=EXYBAR-1.0
      SIGN=-1.0
   30 J=1
      FAC=1.0
      S1=0.0
      S2=0.0
   10 XJ=FLOAT(J)
      ARG=XJ/AB
      G=GAMMA(ARG)
      FAC=(ADL/XJ)*FAC*SIGN
      FF=FAC*G
      FF1=FF*XJ
      IF(ABS(FF1).LE.1.0E-08)GO TO 20
      S1=S1+FF
      S2=S2+FF1
      J=J+1
      GO TO 10
 20   PROD=1.0
      DO 101 I=1,IN
      P=Y(I)*((1./AB)-1.0)
      TRM=EXP(-P)*(1.0+(ADL/AB))
      PROD=PROD*(TRM**(1.0/FLOAT(IN)))
 101  CONTINUE
      DENOM=EXP(S1-0.5772)*PROD
C     ABNXT=(1.0-(1.0/AB))*S2/(DENOM-1.0)
      DNM=AB*(DENOM-1.0)
      ABNXT=1.0/(1.0-DNM/S2)
               WRITE(6,*)AB,ABNXT
   50 J=1
      FAC=1.0
      S1=0.0
      S2=0.0
   60 XJ=FLOAT(J)
      ARG=XJ/ABNXT
      G=GAMMA(ARG)
      FAC=(ADL/XJ)*FAC*SIGN
      FF=FAC*G
      FF1=FF*XJ
      IF(ABS(FF1).LE.1.0E-08)GO TO 70
      S2=S2+FF1
      J=J+1
      GO TO 60
   70 S=0.0
      DO 102 I=1,IN
      EX=EXP(-Y(I)/ABNXT)
      S=S+EX
 102  CONTINUE
      ADLNXT=(-FLOAT(IN)/ABNXT)*S2/S
```

```
                WRITE(6,*)IC,ABNXT,ADLNXT
      IF((ABS(ABNXT-AB).LE.1.0E-05).AND.(ABS(ADLNXT-ADL).LE.1.0E-05))
     *GO TO 40
      AB=ABNXT
      ADL=ADLNXT
      IF(IC.GE.50)GO TO 40
      IC=IC+1
      GO TO 30
   40 AB=ABNXT
      ADL=ADLNXT
      RETURN
      END
C
C-----------       SUBROUTINE YSTAT(  CALLED BY YROOT  )     ----------
C
      SUBROUTINE YSTAT(IN)
      COMMON/YDATA/Y(1700)
      COMMON/YPARM/YBAR,EXYBAR
      S1=0.0
      S2=0.0
      DO 1 I=1,IN
      S1=S1+Y(I)
      S2=S2+EXP(-Y(I))
    1 CONTINUE
      YBAR=S1/FLOAT(IN)
      EXYBAR=S2/FLOAT(IN)
C       WRITE(6,*)IN,(Y(I),I=1,IN)
C       WRITE(6,*)YBAR,EXYBAR
      RETURN
      END
C
C--------     SUBROUTINE FCN (   CALLED BY ZSCNT IN YROOT    )   -------
C
      SUBROUTINE FCN(A,F,N,PAR)
      REAL A(2),F(2),PAR(2)
      COMMON/YDATA/Y(1700)
      IN=1600
      SIGN=-1.0
      FAC=1.0
      S1=0.0
      S2=0.0
      J=1
   10 XJ=FLOAT(J)
      ARG=XJ/A(1)
      G=GAMMA(ARG)
      FAC=(A(2)/XJ)*FAC*SIGN
      FF=FAC*G
      FF1=FF*XJ
      IF(ABS(FF1).LE.1.0E-08)GO TO 20
      S1=S1+FF
      S2=S2+FF1
      J=J+1
      GO TO 10
   20 S3=S2/(A(1)*A(2))
      SA=0.0
      SB=0.0
      DO 30 I=1,IN
      T1=EXP(-Y(I))/A(1)
      T2=ALOG(1.0+(A(2)/A(1))*T1/EXP(-Y(I)))
      SA=SA+T1
```

```
      SB=SB+T2
   30 CONTINUE
      EXPMN=SA/FLOAT(IN)
      ALNMN=SB/FLOAT(IN)
      F(1)=S3+EXPMN
      F(2)=ALOG(1.+(1.-(1./A(1)))*S2/A(1))+0.5772-S1-ALNMN
            WRITE(6,*)F(1),F(2)
      RETURN
      END
C
C-------------------       SUBROUTINE YROOT3      ----------------------
C
      SUBROUTINE YROOT3(AB,ADL)
      DIMENSION WK(42),A(2),PAR(2)
      EXTERNAL FCN3
C     IN=1600
C     CALL YSTAT3(IN,AB,ADL,EXYBAR,TRMR)
      PAR(1)=0.0
      PAR(2)=0.0
      N=2
      NSIG=5
      ITMAX=500
      A(1)=AB
      A(2)=ADL
C     WRITE(6,*)A(1),A(2)
      CALL ZSCNT(FCN3,NSIG,N,ITMAX,PAR,A,FNORM,WK,IER)
C     WRITE(6,*)FNORM,A(1),A(2)
      AB=A(1)
      ADL=A(2)
      RETURN
      END
C
C------------    SUBROUTINE YSTAT3 ( CALLED BY YROOT3 )    -------------
C
      SUBROUTINE YSTAT3(IN,AB,ADL,EXYBAR,TRMR)
      COMMON/YDATA/Y(1700)
      S1=0.0
      S2=0.0
      DO 1 I=1,IN
      A=EXP(-Y(I)/AB)
      S1=S1+A
      B=1.0+(ADL/AB)*(A/EXP(-Y(I)))
      B=ALOG(B)
      S2=S2+B
    1 CONTINUE
C           WRITE(6,*)AB,ADL
      EXYBAR=S1/FLOAT(IN)
      TRMR=S2/FLOAT(IN)
      RETURN
      END
C
C--------     SUBROUTINE FCN3(   CALLED BY ZSCNT IN YROOT3   )   -------
C
      SUBROUTINE FCN3(A,F,N,PAR)
      DIMENSION A(2),F(2),PAR(2)
      AB=A(1)
      ADL=A(2)
      IN=1600
      CALL YSTAT3(IN,AB,ADL,EXYBAR,TRMR)
      PAR(1)=EXYBAR
```

```
      PAR(2)=TRMR
      SIGN=-1.0
      FAC=1.0
      T1=0.0
      J=1
   10 XJ=FLOAT(J)
      ARG=XJ/AB
      G=GAMMA(ARG)
      FAC=(ADL/XJ)*FAC*SIGN
      FF=FAC*G
      FF1=FF*XJ
      IF(ABS(FF1).LE.1.0E-08)GO TO 20
      T1=T1+FF1
      J=J+1
      GO TO 10
C
C
   20 S2=0.0
      S20=0.0
      ICOUNT=1
      DO 2 J=2,1000
      XJ=FLOAT(J)
      ARG=ADL/AB
      A1=(-1.0**XJ)*(ARG**XJ)/(XJ*(XJ-1.0))
C
C
      ARG0=(XJ/AB)-XJ+1.0
      IF(ARG0.LT.0.0)GO TO 12
      B0=GAMMA(ARG0)
      GO TO 13
   12 B0=SGAMMA(ARG0)
C
C
   13 S1=B0
C            WRITE(6,*)J,ARG0,B0,A(1),A(2)
      B1=1.0
      S10=0.0
      K=1
   30 XK=FLOAT(K)
      ARG=((XK+XJ)/AB)-XJ+1.0
      IF(ARG.LT.0.0)GO TO 14
      G=GAMMA(ARG)
      GO TO 16
   14 G=SGAMMA(ARG)
   16 B1=(ADL/XK)*B1*SIGN
      BB1=B1*G
      S1=S1+BB1
      IF(ABS(S1-S10).LE.1.0E-08)GO TO 40
      K=K+1
      S10=S1
      GO TO 30
   40 S2=S2+S1*A1
      IF((ABS(S2-S20).LE.1.0E-08).OR.(ICOUNT.GT.1000))GO TO 50
      S20=S2
      ICOUNT=ICOUNT+1
    2 CONTINUE
C50 WRITE(6,*)J,S2
   50 COMNT=FF1/AB
      F(1)=(COMNT/ADL)+PAR(1)
      F(2)=COMNT*(-1.0)+S2-PAR(2)
```

```
                    WRITE(6,*)ICOUNT,F(1),F(2)
      RETURN
      END
C
C-----------    FUNCTION SGAMA (GAMMA FOR -VE ARGUMENTS)   ------------
C
      REAL FUNCTION SGAMMA(Z)
C     PROD=1.0/Z
C     PROD0=10.0
C     DO 1 N=1,1000
C     XN=FLOAT(N)
C     T=XN/(Z+XN)
C     PROD=PROD*T
C     IF(ABS(PROD-PROD0).LE.1.0E-08)GO TO 10
C     PROD0=PROD
C 1   CONTINUE
C 10  SGAMMA=PROD*(XN**Z)
C
C
      P=1.0/Z
 30   Z1=Z+1
      IF(Z1.GT.0.0)GO TO 20
      P=P/Z1
      Z=Z1
      GO TO 30
 20   SGAMMA=P*GAMMA(Z1)
      RETURN
      END
C
C--------------------       SUBROUTINE YROOT4      ----------------------
C
      SUBROUTINE YROOT4(AB,ADL)
      DIMENSION WK(42),A(2),PAR(2)
      EXTERNAL FCN4
      PAR(1)=0.0
      PAR(2)=0.0
      N=2
      NSIG=5
      ITMAX=500
      A(1)=AB
      A(2)=ADL
C     WRITE(6,*)A(1),A(2)
      CALL ZSCNT(FCN4,NSIG,N,ITMAX,PAR,A,FNORM,WK,IER)
      WRITE(6,*)A(1),A(2),FNORM
      AB=A(1)
      ADL=A(2)
      RETURN
      END
C
C--------     SUBROUTINE FCN4(   CALLED BY ZSCNT IN YROOT4    )   -------
C
      SUBROUTINE FCN4(A,F,N,PAR)
      DIMENSION A(2),F(2),PAR(2)
      AB=A(1)
      ADL=A(2)
      IN=1600
      CALL YSTAT3(IN,AB,ADL,EXYBAR,TRMR)
      PAR(1)=EXYBAR
      PAR(2)=TRMR
      SIGN=-1.0
```

```
      FAC=1.0
      T1=0.0
      J=1
   10 XJ=FLOAT(J)
      ARG=XJ/AB
      G=GAMMA(ARG)
      FAC=(ADL/XJ)*FAC*SIGN
      FF=FAC*G
      FF1=FF*XJ
      IF(ABS(FF1).LE.1.0E-08)GO TO 20
      T1=T1+FF1
      J=J+1
      GO TO 10
C
C
 20   T1=T1/(ADL*AB)
      F(1)=-T1-PAR(1)
C
      PWR=ALOG(3.0)-2*(5.5**(-AB))
      T2A=ADL**PWR
      T2B=(3.0+AB)**2.059
      T2=T2A*T2B/(10.0*EXP(1.0))
      F(2)=T2-PAR(2)
C
C     WRITE(6,*)'  AB = ',AB,'  ADL = ',ADL,'  F(1) = ',F(1),'  F(2) = '
C    *,F(2)
      RETURN
      END
C
C -----------   SUBROUTINE  YROOT5   -----------------
C
      SUBROUTINE YROOT5(AB,ADL)
      COMMON/YPARM/YBAR,EXYBAR
      EXTERNAL FCN5
      IN=1600
      XN=FLOAT(IN)
C
      CALL YSTAT(IN)
C
      EPS=1.0E-5
      NSIG=5
      MAXFN=100
      ABLR=1.55
      ABUR=10.0
      CALL ZBRENT(FCN5,EPS,NSIG,ABLR,ABUR,MAXFN,IER)
C
      AB=ABUR
C
      CALL YSTAT5(IN,AB,EXYAB)
C
      ADL=(1.0-EXYBAR)/EXYAB
      WRITE(6,*)'  AB = ',AB,'  ADL = ',ADL,'  MAXFN = ',MAXFN
      RETURN
      END
C
C -----  FUNCTION SUBPROGRAM FCN5 (CALLED BY ZBRENT IN YROOT5)  --------
C
      REAL FUNCTION FCN5(AB)
      COMMON/YDATA/Y(1700)
      COMMON/YPARM/YBAR,EXYBAR
```

```
      IN=1600
      XN=FLOAT(IN)
      CALL YSTAT5(IN,AB,EXYAB)
C
      YFAC=(1.0-EXYBAR)/EXYAB
      PWR=ALOG(3.0)-2.0*(5.5**(-AB))
      T2A=YFAC**PWR
      T2B=(3.0+AB)**2.059
      T=T2A*T2B/(10.0*EXP(1.0))
C
      S=0.0
      DO 1 I=1,IN
      A1=EXP(-Y(I)*((1.0/AB)-1.0))
      ARG=1.0+(YFAC/AB)*A1
      S=S+ALOG(ARG)
    1 CONTINUE
      SLOGAV=S/XN
      FCN5=T-SLOGAV
      RETURN
      END
C
C --------   SUBROUTINE YSTAT5 (CALLED BY FCN5 & YROOT5)  --------------
C
      SUBROUTINE YSTAT5(IN,AB,EXYAB)
      COMMON/YDATA/Y(1700)
      S=0.0
      DO 1 I=1,IN
      S=S+EXP(-Y(I)/AB)
    1 CONTINUE
      EXYAB=S/FLOAT(IN)
      RETURN
      END
C
C -----------   SUBROUTINE  YROOT6   -----------------
C
      SUBROUTINE YROOT6(AB,ADL)
      COMMON/YPARM/YBAR,EXYBAR
      EXTERNAL FCN6
      IN=1600
      XN=FLOAT(IN)
      CALL YSTAT(IN)
      EPS=1.0E-5
      NSIG=5
      MAXFN=100
      ABLR=0.5*AB
      ABUR=1.5*AB
      CALL ZBRENT(FCN6,EPS,NSIG,ABLR,ABUR,MAXFN,IER)
C
      CALL YSTAT5(IN,AB,EXYAB)
      ADL=(1.0-EXYBAR)/EXYAB
      AB=ABUR
      WRITE(6,*)'  AB = ',AB,'  ADL = ',ADL,'  MAXFN = ',MAXFN
      RETURN
      END
C
C -----  FUNCTION SUBPROGRAM FCN6 (CALLED BY ZBRENT IN YROOT6)  --------
C
      REAL FUNCTION FCN6(AB)
      COMMON/YDATA/Y(1700)
      COMMON/YPARM/YBAR,EXYBAR
```

```
      IN=1600
      XN=FLOAT(IN)
C
      CALL YSTAT5(IN,AB,EXYAB)
      YFAC=(1.0-EXYBAR)/EXYAB
C
      PWR=ALOG(3.0)-2.0*(5.5**(-AB))
      T2A=YFAC**PWR
      T2B=(3.0+AB)**2.059
      T=T2A*T2B/(10.0*EXP(1.0))
C
      CALL SIGMA(AB,YFAC,A,B)
      TT=A-0.57721
C
      S=0.0
      DO 1 I=1,IN
      ARG=EXP(-Y(I))+(YFAC/AB)*EXP(-Y(I)/AB)
      S=S+ALOG(ARG)
    1 CONTINUE
      SLOGAV=S/XN
C
      FCN6=T+TT-SLOGAV
C
      RETURN
      END
C
C -------------    SUBROUTINE  SIGMA     ------------------------------
C
      SUBROUTINE SIGMA(AB,ADL,A,B)
      SIGN=-1.0
      A=0.0
      B=0.0
      F=1.0
      J=1
   10 XJ=FLOAT(J)
      ARG=XJ/AB
      G=GAMMA(ARG)
      F=(ADL/XJ)*F*SIGN
      FF=F*G
      FF1=FF*XJ
      IF(ABS(FF1).LE.1.0E-08)GO TO 20
      A=A+FF
      B=B+FF1
      J=J+1
      GO TO 10
   20 RETURN
      END
C
C -------    SUBROUTINE ESTMLE (MLE ESTIMATE OF AB1 AND ADL1)   --------
C
      SUBROUTINE ESTMLE(AB1,ADL1,IFL)
      COMMON/PARA/X(50,50),C
      COMMON/FLAG/IFLAG
      EXTERNAL FXVAR
      IFLAG=IFL
      EPS=1.0E-05
      NSIG=5
      MAXFN=100
      AB1LR=2.00
      AB1UR=40.00
```

```
      CALL ZBRENT(FXVAR,EPS,NSIG,AB1LR,AB1UR,MAXFN,IER)
      AB1=AB1UR
C
      S=0.0
      DO 10 J=1,40
      S=S+EXP(-X(IFL,J)/AB1)
 10   CONTINUE
      ADL1=40.00/S
      WRITE(6,*)'  AB1 = ',AB1,'  ADL1 = ',ADL1
      RETURN
      END
C
      REAL FUNCTION FXVAR(AB1)
      COMMON/PARA/X(50,50),C
      COMMON/FLAG/IFLAG
      COMMON/XSTAT/XAVG(40)
C
      S1=0.0
      S2=0.0
      DO 10 J=1,40
      T=EXP(-X(IFLAG,J)/AB1)
      S1=S1+T
      S2=S2+X(IFLAG,J)*T
 10   CONTINUE
      FXVAR=S2-(XAVG(IFLAG)-AB1)*S1
      RETURN
      END
$ENTRY
//GO.FT08F001 DD DSN=CEAROR.TCEV.LIER1,DISP=SHR
//GO.FT09F001 DD DSN=CEAROR.TCEV.BASC1,DISP=SHR
$$
//
```

```
//PROJECT JOB (1304,59634,2,20),'ARORA',MSGCLASS=S,CLASS=A
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=N
//     EXEC WATFIV,REGION.GO=4000K,TIME.GO=99
$JOB            TIME=4500
C=====================================================================
C              TWO COMPONENT EXTREME VALUE DISTRIBUTION
C---------------------------------------------------------------------
C  THIS PROGRAM COMPUTES THE STATISTICAL PROPERTIES OF SEVERAL ENTROPY
C  ESTIMATORS OF TCEV WHOSE VALUES WERE COMPUTED BY TCEV.PROG(REGION),
C  THE REGIONAL ESTIMATION CODE.
C=====================================================================
      REAL AB1(40),ADL1(40),AB2(40),ADL2(40)
      DIMENSION FPR(4),PPQN(4)
      REAL PPPSQN(4),PPEXQN(4)
      REAL SQN(4),S2QN(4),SPSQN(4),S2PSQN(4),SEXQN(4),S2EXQN(4)
      REAL MAB,MADL,MSAB,MSADL,M2AB,M2ADL
      REAL MAB1,MADL1,MSAB1,MSADL1,MC1,MSC1,M2C1,M2AB1,M2ADL1
      REAL MQN(4),MSQN(4),MPSQN(4),MSPSQN(4),MEXQN(4),MSEXQN(4)
      REAL M2QN(4),M2PSQN(4),M2EXQN(4)
      REAL BQN(4),BPSQN(4),BEXQN(4)
      REAL VQN(4),VPSQN(4),VEXQN(4)
C     REAL MXAB1,MNAB1,MXADL1,MNADL1
      DATA FPR(1),FPR(2),FPR(3),FPR(4)/0.5,0.9,0.99,0.999/
      DATA PPQN(1),PPQN(2),PPQN(3),PPQN(4)/28.99104,54.43663,
     *110.7783,181.0672/
C=====================================================================
      NREP=100
C
      SAB=0.0
      SADL=0.0
      S2AB=0.0
      S2ADL=0.0
      SAB1=0.0
      SADL1=0.0
      S2AB1=0.0
      S2ADL1=0.0
      SC1=0.0
      S2C1=0.0
C
      DO 11 I=1,4
      SQN(I)=0.0
      S2QN(I)=0.0
      SPSQN(I)=0.0
      S2PSQN(I)=0.0
      SEXQN(I)=0.0
      S2EXQN(I)=0.0
  11  CONTINUE
C
      PPSI=10.0*ALOG(10.0)
           CALL SIGMA(3.067,0.173,A,B)
           CALL EXPEC(10.0,10.0,A,PPSI,PEX)
      DO 12 J=1,4
      PPPSQN(J)=PPQN(J)/PPSI
```

```
  12  PPEXQN(J)=PPQN(J)/PEX
C
C=============================================================
C
      DO 10 INR=1,NREP
C
      IF(INR.LE.93)THEN
      READ(8,*)AB,ADL
      ELSE
      READ(9,*)AB,ADL
      END IF
            CALL SIGMA(AB,ADL,AA,B)
      SAB=SAB+AB
      S2AB=S2AB+AB*AB
      SADL=SADL+ADL
      S2ADL=S2ADL+ADL*ADL
C
      IF(INR.LE.93)THEN
      READ(10,*)(AB1(I),I=1,40)
      READ(10,*)(ADL1(I),I=1,40)
      ELSE
      READ(11,*)(AB1(I),I=1,40)
      READ(11,*)(ADL1(I),I=1,40)
      END IF
C
      DO 1 I=1,40
C
      ADL2(I)=ADL*(ADL1(I)**(1.0/AB))
      AB2(I)= AB*AB1(I)
C
      SAB1=SAB1+AB1(I)
      S2AB1=S2AB1+AB1(I)*AB1(I)
      SADL1=SADL1+ADL1(I)
      S2ADL1=S2ADL1+ADL1(I)*ADL1(I)
C
      C1=0.557/(ALOG10(ADL1(I))+0.251)
      SC1=SC1+C1
      S2C1=S2C1+C1*C1
C
      PSI=AB1(I)*ALOG(ADL1(I))
      CALL EXPEC(AB1(I),ADL1(I),AA,PSI,EX)
C
           DO 2 J=1,4
           FF=FPR(J)
           CALL TCEVVA(ADL1(I),ADL2(I),AB1(I),AB2(I),FF,XX)
           XQN=XX
           SQN(J)=SQN(J)+XQN
           S2QN(J)=S2QN(J)+XQN*XQN
C
           XPSQN=XQN/PSI
           SPSQN(J)=SPSQN(J)+XPSQN
           S2PSQN(J)=S2PSQN(J)+XPSQN*XPSQN
C
           XEXQN=XQN/EX
           SEXQN(J)=SEXQN(J)+XEXQN
           S2EXQN(J)=S2EXQN(J)+XEXQN*XEXQN
  2        CONTINUE
C
  1     CONTINUE
C
```

```
  10  CONTINUE
C
C ----------------      STATISTICAL PROPERTIES      ------------------
C
C.......................AB AND ADL.........................
C
      XREP=FLOAT(NREP)
      MAB=SAB/XREP
      MADL=SADL/XREP
      BAB=MAB-3.067
      BADL=MADL-0.173
C
      M2AB=S2AB/XREP
      M2ADL=S2ADL/XREP
      VAB=(M2AB-MAB*MAB)*XREP/(XREP-1.0)
      VADL=(M2ADL-MADL*MADL)*XREP/(XREP-1.0)
      MSAB=((BAB*BAB+VAB)**0.5)/3.067
      MSADL=((BADL*BADL+VADL)**0.5)/0.173
      BAB=BAB/3.067
      BADL=BADL/0.173
      WRITE(6,*)'*** OUTLIER COMPONENT PARAMETERS ***'
      WRITE(6,*)'  NO. OF SAMPLES = ',NREP
      WRITE(6,*)'  BIAS(AB) = ',BAB,'  BIAS(ADL) = ',BADL
      WRITE(6,*)'  VAR(AB) = ',VAB,'  VAR(ADL) = ',VADL
      WRITE(6,*)'  MSE(AB) = ',MSAB,'  MSE(ADL) = ',MSADL
C
C.......................AB1, ADL1 AND C1....................
C
      XREP1=FLOAT(40*NREP)
      MAB1=SAB1/XREP1
      MADL1=SADL1/XREP1
      MC1=SC1/XREP1
      BAB1=MAB1-10.00
      BADL1=MADL1-10.00
           PC1=0.557/(ALOG10(10.00)+0.251)
      BC1=MC1-PC1
C
      M2AB1=S2AB1/XREP1
      M2ADL1=S2ADL1/XREP1
      M2C1=S2C1/XREP1
      VAB1=(M2AB1-MAB1*MAB1)*XREP1/(XREP1-1.0)
      VADL1=(M2ADL1-MADL1*MADL1)*XREP1/(XREP1-1.0)
      VC1=(M2C1-MC1*MC1)*XREP1/(XREP1-1.0)
      MSAB1=((BAB1*BAB1+VAB1)**0.5)/10.0
      MSADL1=((BADL1*BADL1+VADL1)**0.5)/10.0
      MSC1=((BC1*BC1+VC1)**0.5)/10.0
      BAB1=BAB1/10.0
      BADL1=BADL1/10.0
      BC1=BC1/PC1
      WRITE(6,*)'    '
      WRITE(6,*)'    '
      WRITE(6,*)'***************  BASIC COMPONENT PARAMETERS  **********'
      WRITE(6,*)'  NO. OF SAMPLES = ',XREP1
      WRITE(6,*)'  BIAS(AB1) = ',BAB1,'  BIAS(ADL1) = ',BADL1
      WRITE(6,*)'  VAR(AB1) = ',VAB1,'  VAR(ADL1) = ',VADL1
      WRITE(6,*)'  MSE(AB1) = ',MSAB1,'  MSE(ADL1) = ',MSADL1
      WRITE(6,*)'    '
      WRITE(6,*)'    '
      WRITE(6,*)'***************  VARIABLE     C1     ******************'
      WRITE(6,*)'  NO. OF SAMPLES = ',XREP1
```

```
      WRITE(6,*)'  BIAS(C1) = ',BC1
      WRITE(6,*)'  VAR(C1) = ',VC1
      WRITE(6,*)'  MSE(C1) = ',MSC1
C
C.....QUANTILES.....
C
          DO 3 J=1,4
          MQN(J)=SQN(J)/XREP1
          M2QN(J)=S2QN(J)/XREP1
          VQN(J)=(M2QN(J)-MQN(J)*MQN(J))*XREP1/(XREP1-1.0)
          BQN(J)=MQN(J)-PPQN(J)
          MSQN(J)=((BQN(J)*BQN(J)+VQN(J))**0.5)/PPQN(J)
          BQN(J)=BQN(J)/PPQN(J)
   3      CONTINUE
C
          WRITE(6,*)'   '
          WRITE(6,*)'   '
          WRITE(6,*)'****************** QUANTILES ******************'
          WRITE(6,*)'   '
          WRITE(6,*)' PROB.  ',(FPR(I),I=1,4)
          WRITE(6,*)'   '
          WRITE(6,*)' BIAS   ',(BQN(I),I=1,4)
          WRITE(6,*)' VAR.   ',(VQN(I),I=1,4)
          WRITE(6,*)' MSE    ',(MSQN(I),I=1,4)
C
          DO 4 J=1,4
          MPSQN(J)=SPSQN(J)/XREP1
          M2PSQN(J)=S2PSQN(J)/XREP1
          VPSQN(J)=(M2PSQN(J)-MPSQN(J)*MPSQN(J))*XREP1/(XREP1-1.0)
          BPSQN(J)=MPSQN(J)-PPPSQN(J)
          MSPSQN(J)=((BPSQN(J)*BPSQN(J)+VPSQN(J))**0.5)/PPPSQN(J)
          BPSQN(J)=BPSQN(J)/PPPSQN(J)
C
          MEXQN(J)=SEXQN(J)/XREP1
          M2EXQN(J)=S2EXQN(J)/XREP1
          VEXQN(J)=(M2EXQN(J)-MEXQN(J)*MEXQN(J))*XREP1/(XREP1-1.0)
          BEXQN(J)=MEXQN(J)-PPEXQN(J)
          MSEXQN(J)=((BEXQN(J)*BEXQN(J)+VEXQN(J))**0.5)/PPEXQN(J)
          BEXQN(J)=BEXQN(J)/PPEXQN(J)
   4      CONTINUE
C
          WRITE(6,*)'   '
          WRITE(6,*)'   '
          WRITE(6,*)'*************** QUANT/EPS ******************'
          WRITE(6,*)'   '
          WRITE(6,*)' PROB.  ',(FPR(I),I=1,4)
          WRITE(6,*)'   '
          WRITE(6,*)' BIAS   ',(BPSQN(I),I=1,4)
          WRITE(6,*)' VAR.   ',(VPSQN(I),I=1,4)
          WRITE(6,*)' MSE    ',(MSPSQN(I),I=1,4)
C
          WRITE(6,*)'   '
          WRITE(6,*)'   '
          WRITE(6,*)'*************** QUANT/EX ******************'
          WRITE(6,*)'   '
          WRITE(6,*)' PROB.  ',(FPR(I),I=1,4)
          WRITE(6,*)'   '
          WRITE(6,*)' BIAS   ',(BEXQN(I),I=1,4)
          WRITE(6,*)' VAR.   ',(VEXQN(I),I=1,4)
          WRITE(6,*)' MSE    ',(MSEXQN(I),I=1,4)
```

```
C
      STOP
      END
C
C
      SUBROUTINE EXPEC(AB1,ADL1,A,PSI,EX)
      EX=AB1*(0.57721-A)+PSI
      RETURN
      END
C
C
      SUBROUTINE TCEVVA(ADL1,ADL2,AB1,AB2,FF,XX)
C=====================================================
C  SUBROUTINE PER IL CALCOLO DI X UNA VOLTA ASSEGNATO
C  UN VALORE DELLA F(X) DEL MODELLO TCEV
C=====================================================
      REAL*8 DL1,DL2,B1,B2,F,X,FX
      COMMON DL1,DL2,B1,B2
      DL1=ADL1
      DL2=ADL2
      B1=AB1
      B2=AB2
      F=FF
      X=B1*DLOG(DL1)-B1*ALOG(ALOG(1/.35))
10    FR=FX(X)
      IF(DABS(FR-F).LT.0.0000001) THEN
      GOTO 57
      ELSE
      X=X-(FR-F)/F1X(X)
      END IF
      GOTO 10
 57   CONTINUE
      XX=X
      RETURN
      END
C
      FUNCTION FX(X)
      REAL*8 FX
      REAL*8 DL1,DL2,B1,B2,X
      COMMON DL1,DL2,B1,B2
      FX=DEXP(-DL1*DEXP(-X/B1)-DL2*DEXP(-X/B2))
C     WRITE(6,*)FX
      RETURN
      END
C
      REAL FUNCTION F1X(X)
      REAL*8 DL1,DL2,B1,B2,X,FX
      COMMON DL1,DL2,B1,B2
      F1X=FX(X)*((DL1/B1)*DEXP(-X/B1)+(DL2/B2)*DEXP(-X/B2))
      RETURN
      END
C
C ------------    SUBROUTINE  SIGMA     ------------------------------
C
      SUBROUTINE SIGMA(AB,ADL,A,B)
      SIGN=-1.0
      A=0.0
      B=0.0
      F=1.0
      J=1
```

```
   10  XJ=FLOAT(J)
       ARG=XJ/AB
       G=GAMMA(ARG)
       F=(ADL/XJ)*F*SIGN
       FF=F*G
       FF1=FF*XJ
       IF(ABS(FF1).LE.1.0E-08)GO TO 20
       A=A+FF
       B=B+FF1
       J=J+1
       GO TO 10
   20  RETURN
       END
$ENTRY
//GO.FT08F001 DD DSN=CEAROR.TCEV.LIER,DISP=SHR
//GO.FT09F001 DD DSN=CEAROR.TCEV.LIER1,DISP=SHR
//GO.FT10F001 DD DSN=CEAROR.TCEV.BASC,DISP=SHR
//GO.FT11F001 DD DSN=CEAROR.TCEV.BASC1,DISP=SHR
$$
//
```

```
//PROJECT JOB (1304,77493,6,20),'ARORA',MSGCLASS=S,CLASS=A
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=N
//    EXEC WATFIV
$JOB           TIME=120
C
C=============================================================================
C              TWO COMPONENT EXTREME VALUE DISTRIBUTION
C-----------------------------------------------------------------------------
C     THIS PROGRAM COMPUTES THE 4 TCEV PARAMETERS FOR A SINGLE SERIES
C             .... COPIED FROM FLOOD.B(REGION) AND MODIFIED ....
C                                                          NOV 23,1985
C=============================================================================
C
      DIMENSION AB1(1),ADL1(1)
      COMMON/PARA/X(1,34),C
      COMMON/YDATA/Y(34)
      COMMON/XSTAT/XAVG(1)
      COMMON/CONST/A1
      COMMON/SAMPC/NSAMPL
C
C------------------      INITIALIZATION      ----------------------------
C
      AB1(1)=120.5
      ADL1(1)=4.2
C
      ABP=0.0
      ADLP=0.0
      AB=6.22
      ADL=0.22
C
C---------------------     READ SINGLE SERIES     ----------------------
C
      READ(5,*)(X(1,J),J=1,34)
C
      WRITE(6,*)'   DATA X(1,J) : ',(X(1,J),J=1,34)
C
C---------------          SAMPLE STATISTICS        ---------------------
C
      XAVG(1)=0.0
      DO 5 J=1,34
  5   XAVG(1)=XAVG(1)+X(1,J)
      XAVG(1)=XAVG(1)/34.0
C
      WRITE(6,*)'   XAVG(1) = ',XAVG(1)
C
C---------      TRANSFORMATION OF ALL SAMPLES TO POOLED DATA    ---------
C
      NIT=1
 222  ICOUNT=1
      DO 33 J=1,34
      Y(ICOUNT)=(X(1,J)/AB1(1))-ALOG(ADL1(1))
      ICOUNT=ICOUNT+1
  33  CONTINUE
```

```
C
C     WRITE(6,*)'      '
C     WRITE(6,*)'      '
C     WRITE(6,*)'    ITERATION NO. : ',NIT
C     WRITE(6,*)'   ------------------------------'
C     WRITE(6,*)'      '
C     WRITE(6,*)'    POOLED DATA Y(I) : ',(Y(I),I=1,34)
C
C-------        PARAMETERS OF POOLED SAMPLE (EQNS (36),(37))    --------
C
      CALL YROOT4(AB,ADL)
C
C     IF(NIT.GT.10)GO TO 999
C
      IF((ABS(AB-ABP).LE.1.0E-05).AND.(ABS(ADL-ADLP).LE.1.E-05))
     *GO TO 999
      NIT=NIT+1
      ABP=AB
      ADLP=ADL
C
C-----     CALCULATION OF CONSTANT C=C(AB,ADL) IN THE FUNCTION
C             F(AB1(I))=0, AND A1=A1(AB,ADL) IN G(ADL1(I)=0    --------
C
      SIGN=-1.0
      A=0.0
      B=0.0
      F=1.0
      J=1
  10  XJ=FLOAT(J)
      ARG=XJ/AB
      G=GAMMA(ARG)
      F=(ADL/XJ)*F*SIGN
      FF=F*G
      FF1=FF*XJ
      IF(ABS(FF1).LE.1.0E-08)GO TO 20
      A=A+FF
      B=B+FF1
      J=J+1
      GO TO 10
  20  A1=40.0*(1.+B/AB)
      AA=ALOG(A1)
      C=AA+0.5772-A
C     WRITE(6,*)'  A1 = ',A1,'  C = ',C
C
C-------------        ENTROPY PARAMETER ESTIMATION
C                        AND THEIR SUMMATIONS     ------------------
C
      NSAMPL=1
      APAB11=50.0
      APAB1=200.0
      CALL ROOT(APAB1,APAB11,APADL1)
      AB1(NSAMPL)=APAB1
      ADL1(NSAMPL)=APADL1
      WRITE(6,*)'              AB1 = ',AB1(1)
      WRITE(6,*)'             ADL1 = ',ADL1(1)
              GO TO 222
 999  WRITE(6,*)'  NO. OF ITERATIONS = ',NIT
      WRITE(6,*)' ----------------------------------'
      STOP
      END
```

```
C=====================================================================
C                        END OF MAIN PROGRAM
C=====================================================================
C:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C=====================================================================
C
C-------------------        SUBROUTINE YROOT4      ----------------------
C
      SUBROUTINE YROOT4(AB,ADL)
      DIMENSION WK(42),A(2),PAR(2)
      EXTERNAL FCN4
      PAR(1)=0.0
      PAR(2)=0.0
      N=2
      NSIG=5
      ITMAX=500
      A(1)=AB
      A(2)=ADL
      CALL ZSCNT(FCN4,NSIG,N,ITMAX,PAR,A,FNORM,WK,IER)
C
      WRITE(6,*)'  AB = ',A(1),'  ADL = ',A(2),'  FNORM = ',FNORM
C
      AB=A(1)
      ADL=A(2)
      RETURN
      END
C
C--------     SUBROUTINE FCN4(   CALLED BY ZSCNT IN YROOT4    )   -------
C
      SUBROUTINE FCN4(A,F,N,PAR)
      DIMENSION A(2),F(2),PAR(2)
      AB=A(1)
      ADL=A(2)
      IN=34
      CALL YSTAT4(IN,AB,ADL,EXYBAR,TRMR)
      PAR(1)=EXYBAR
      PAR(2)=TRMR
      SIGN=-1.0
      FAC=1.0
      T1=0.0
      J=1
  10  XJ=FLOAT(J)
      ARG=XJ/AB
      G=GAMMA(ARG)
      FAC=(ADL/XJ)*FAC*SIGN
      FF=FAC*G
      FF1=FF*XJ
      IF(ABS(FF1).LE.1.0E-08)GO TO 20
      T1=T1+FF1
      J=J+1
      GO TO 10
C
C
 20   T1=T1/(ADL*AB)
      F(1)=-T1-PAR(1)
C
      PWR=ALOG(3.0)-2*(5.5**(-AB))
      T2A=ADL**PWR
      T2B=(3.0+AB)**2.059
```

```
      T2=T2A*T2B/(10.0*EXP(1.0))
      F(2)=T2-PAR(2)
C
C     WRITE(6,*)'  AB = ',AB,'  ADL = ',ADL
C     WRITE(6,*)'  F(1) = ',F(1),'  F(2) = ',F(2)
C     WRITE(6,*)'          '
C
      RETURN
      END
C
C------------     SUBROUTINE YSTAT4 ( CALLED BY FCN4 )     ------------
C
      SUBROUTINE YSTAT4(IN,AB,ADL,EXYBAR,TRMR)
      COMMON/YDATA/Y(34)
      S1=0.0
      S2=0.0
      DO 1 I=1,IN
      A=EXP(-Y(I)/AB)
      S1=S1+A
      B=1.0+(ADL/AB)*(A/EXP(-Y(I)))
      B=ALOG(B)
      S2=S2+B
  1   CONTINUE
      EXYBAR=S1/FLOAT(IN)
      TRMR=S2/FLOAT(IN)
C
C     WRITE(6,*)'  AB = ',AB,'  ADL = ',ADL,'  EXYBAR = ',EXYBAR,
C    *'  TRMR   = ',TRMR
C
      RETURN
      END
C
C-------------------        SUBROUTINE ROOT      ----------------------
C
      SUBROUTINE ROOT(AB1,AB11,ADL1)
      COMMON/CONST/A1
      EXTERNAL FF
C
C---------------     ITERATIVE ESTIMATION OF AB1
C                    USING IMSL ROUTINE ZBRENT    -------------------
C
      EPS=1.0E-5
      NSIG=5
      MAXFN=100
      CALL ZBRENT(FF,EPS,NSIG,AB11,AB1,MAXFN,IER)
      ADL1=A1/ESUM(AB1)
C
C         WRITE(6,*)'  AB1 = ',AB1,'  ADL1 = ',ADL1
C         WRITE(6,*)'          '
C
      RETURN
      END
C
      REAL FUNCTION FF(THETA)
      COMMON/PARA/X(1,34),C
      COMMON/XSTAT/XAVG(1)
      COMMON/SAMPC/NSAMPL
      S=0.0
      DO 10 I=1,34
      TERM=EXP(-X(NSAMPL,I)/THETA)
```

```
      S=S+TERM
   10 CONTINUE
      S=ALOG(S)
      FF=XAVG(NSAMPL)+THETA*(S-C)
C
C     WRITE(6,*)'  AB1 = ',THETA,'  FF = ',FF
C
      RETURN
      END
C
      REAL FUNCTION ESUM(XX)
      COMMON/PARA/X(1,34),C
      COMMON/SAMPC/NSAMPL
      S=0.0
      DO 10 I=1,34
      TERM=EXP(-X(NSAMPL,I)/XX)
      S=S+TERM
   10 CONTINUE
      ESUM=S
      RETURN
      END
$ENTRY
135. 150. 78. 448. 368. 87. 1590. 345. 320. 162. 216. 680. 88.8 70.
2300. 250. 454. 30.3 272. 137. 1064. 552. 267. 219. 484. 66.2 168.
253. 417. 265. 450. 307. 117. 660.
$$
//
```

```
//PROJECT JOB (1304,77493,6,20),'ARORA',MSGCLASS=S,CLASS=Q
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=D
//     EXEC WATFIV
$JOB            TIME=60
C
C=================================================================
C               TWO COMPONENT EXTREME VALUE DISTRIBUTION
C -----------------------------------------------------------------
C       THIS PROGRAM SOLVES THE FOUR SIMULTANEOUS TCEV EQUATIONS
C       IN X - VARIATE...
C                ... COPIED FROM FLOOD.TCEV(SSER), THE CODE TO COMPUTE
C                 THE TCEV PARAMETERS OF A SINGLE SERIES, AND MODIFIED...
C                                                        NOV 24,1985
C=================================================================
C
      COMMON/XDATA/X(34)
      COMMON/XST/AVGX
C
C------------------      INITIALIZATION      ----------------------------
C
      AB1=120.5
      ADL1=4.2
      AB=6.22
      ADL=0.22
C
C----------------------     READ SINGLE SERIES     ----------------------
C
      READ(5,*)(X(J),J=1,34)
C
      WRITE(6,*)'   DATA X(J) : ',(X(J),J=1,34)
C
C---------------          SAMPLE STATISTICS        ----------------------
C
      AVGX=0.0
      DO 5 I=1,34
  5   AVGX=AVGX+X(I)
      AVGX=AVGX/34.0
C
      WRITE(6,*)'   AVGX = ',AVGX
C
C--------       ROOTS OF THE SIMULTANEOUS EQUATIONS IN X      ------------
C
      CALL XROOT(AB1,ADL1,AB,ADL)
C
      WRITE(6,*)'  AB1 = ',AB1,'  ADL1 = ',ADL1,'  AB = ',AB,'  ADL = ',
     *ADL
C
 999  STOP
      END
C=================================================================
C                          END OF MAIN PROGRAM
C=================================================================
C:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
```

```
C::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C==========================================================================
C
C------------------        SUBROUTINE XROOT      ------------------------
C
      SUBROUTINE XROOT(AB1,ADL1,AB,ADL)
      DIMENSION WK(100),A(4),PAR(4)
      EXTERNAL FCN
      PAR(1)=0.0
      PAR(2)=0.0
      PAR(3)=0.0
      PAR(4)=0.0
      N=4
      NSIG=5
      ITMAX=500
      A(1)=AB1
      A(2)=ADL1
      A(3)=AB
      A(4)=ADL
      CALL ZSCNT(FCN,NSIG,N,ITMAX,PAR,A,FNORM,WK,IER)
C
      WRITE(6,*)'  AB1 = ',A(1),'  ADL1 = ',A(2)
      WRITE(6,*)'  AB = ',A(3),'  ADL = ',A(4),'  FNORM = ',FNORM
C
      AB1=A(1)
      ADL1=A(2)
      AB=A(3)
      ADL=A(4)
      RETURN
      END
C
C--------      SUBROUTINE FCN(   CALLED BY ZSCNT IN XROOT   )   ---------
C
      SUBROUTINE FCN(A,F,N,PAR)
      COMMON/XST/AVGX
      DIMENSION A(4),F(4),PAR(4)
C
      AB1=A(1)
      ADL1=A(2)
      AB=A(3)
      ADL=A(4)
      IN=34
      CALL XSTAT(IN,AB1,ADL1,AB,ADL,AVGEX,AVGEX1,AVGLNX)
      PAR(1)=AVGX
      PAR(2)=AVGEX
      PAR(3)=AVGEX1
      PAR(4)=AVGLNX
C
      SIGN=-1.0
      FAC=1.0
      T1=0.0
      T2=0.0
      J=1
   10 XJ=FLOAT(J)
      ARG=XJ/AB
      G=GAMMA(ARG)
      FAC=(ADL/XJ)*FAC*SIGN
      FF=FAC*G
      FF1=FF*XJ
      IF(ABS(FF1).LE.1.0E-08)GO TO 20
```

```
      T1=T1+FF
      T2=T2+FF1
      J=J+1
      GO TO 10
C
C
 20   F(1)=AB1*(ALOG(ADL1)+0.57721-T1)-PAR(1)
      F(2)=(1.0+T2/AB)/ADL1-PAR(2)
      F(3)=T2/(AB*ADL*(ADL1**(1.0/AB)))+PAR(3)
C
      PWR=ALOG(3.0)-2*(5.5**(-AB))
      TA=ADL**PWR
      TB=(3.0+AB)**2.059
      TT=TA*TB/(10.0*EXP(1.0))
C
      F(4)=TT-PAR(4)
C
      WRITE(6,*)'  AB1 = ',AB1,'  ADL1 = ',ADL1,'  AB = ',AB,'  ADL = ',
     *ADL
      WRITE(6,*)'  F(1) = ',F(1),'  F(2) = ',F(2),'  F(3) = ',F(3),
     *'  F(4) = ',F(4)
      WRITE(6,*)'          '
C
      RETURN
      END
C
C------------    SUBROUTINE XSTAT ( CALLED BY FCN )    -------------
C
      SUBROUTINE XSTAT(IN,AB1,ADL1,AB,ADL,AVGEX,AVGEX1,AVGLNX)
      COMMON/XDATA/X(34)
      S1=0.0
      S2=0.0
      S3=0.0
      S4=0.0
      DO 1 I=1,IN
      A=EXP(-X(I)/AB1)
      B=EXP(-X(I)/(AB*AB1))
      XPNT=(1.0/AB-1.0)
      EXPNT=(X(I)*XPNT)/AB1
      ARG=1.0+(ADL/AB)*(ADL1**XPNT)*EXP(-EXPNT)
      C=ALOG(ARG)
      S1=S1+A
      S2=S2+B
      S3=S3+C
  1   CONTINUE
C
      XN=FLOAT(IN)
      AVGEX=S1/XN
      AVGEX1=S2/XN
      AVGLNX=S3/XN
C
C     WRITE(6,*)'  AB = ',AB,'  ADL = ',ADL,'  EXYBAR = ',EXYBAR,
C    *'  TRMR   = ',TRMR
C
      RETURN
      END
$ENTRY
135. 150. 78. 448. 368. 87. 1590. 345. 320. 162. 216. 680. 88.8 70.
2300. 250. 454. 30.3 272. 137. 1064. 552. 267. 219. 484. 66.2 168.
253. 417. 265. 450. 307. 117. 660.
```

```
$$
//
//*//GO.SYSIN DD *
//*GO.FT08F001 DD DSN=CEAROR.OUT.G21,DISP=SHR
//*GO.FT09F001 DD DSN=CEAROR.OUT.G22,DISP=SHR
```

```
//PROJECT JOB (1304,77493,6,20),'ARORA',MSGCLASS=S,CLASS=A
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=D
//      EXEC WATFIV,REGION.GO=4000K,TIME.GO=200
$JOB            TIME=60
C
C=================================================================
C               TWO COMPONENT EXTREME VALUE DISTRIBUTION
C-----------------------------------------------------------------
C       NUMERICAL CALCULATION OF THE INTEGRAL FOR THE LAST CONSTRAINT
C       IN THE ENTROPY FORMULATION.
C=================================================================
        REAL LNADL
        DOUBLE PRECISION DSUM,DAB,DAREA
        DIMENSION T(15,15),ADLL(10)
C
        DEL=0.5
        AB=1.5
C
        DO 10 I=1,14
        WRITE(6,*)'        '
        WRITE(6,*)'  AB = ',AB
        WRITE(6,*)'  --------------------------'
        WRITE(6,*)'        '
        LNADL=-5.0
        ADL=EXP(LNADL)
        DO 20 J=1,11
        CALL SIGMA(AB,ADL,DSUM)
        CALL INTGRL(AB,ADL,AREA)
        DAREA=AREA
        TSUM=-DAREA-DSUM
C       T(I,J)=TSUM
        WRITE(6,*)'  LNADL = ',LNADL,'  ADL = ',ADL,'  SUM = ',TSUM
        WRITE(6,*)'        '
        LNADL=LNADL+DEL
        ADL=EXP(LNADL)
  20    CONTINUE
        AB=AB+DEL
  10    CONTINUE
C
C       LNADL=5.0
C       DEL=-0.5
C       DO 25 J=1,11
C       WRITE(9,51)LNADL,(T(I,J),I=1,7)
C       WRITE(10,51)LNADL,(T(I,J),I=8,14)
C       LNADL=LNADL+DEL
C 25    CONTINUE
C 51    FORMAT(1X,F8.3,7(F13.7,1X))
C-----------------------------------------------------------------
C
C-----------    CALCULATION OF P2(ADL) = SIGMA(.)/(J-1)!   ------------
C
C       READ(5,*)(ADLL(I),I=1,10)
C       DO 10 I=1,10
```

```
C       ADL=ADLL(I)
C       WRITE(6,*)'                    '
C       WRITE(6,*)'                    '
C       WRITE(6,*)'  ADL = ',ADL
C       WRITE(6,*)' ----------------------------'
C       AB=1.0
C 20    CALL SIGMA(AB,ADL,DSUM)
C       DAB=AB
C       DSUM=DSUM/DAB
C       WRITE(6,*)'  AB = ',AB,'  P2(ADL) = ',DSUM
C       IF(AB.EQ.5.00)GO TO 10
C       AB=5.0
C       GO TO 20
C 10    CONTINUE
        STOP
        END
C
C
        SUBROUTINE INTGRL(AB,ADL,AREA)
        RATIO=ADL/AB
        YDEL=0.0001
        AREA=0.0
        Y=0.0
        Z0=0.0
        FZ0=ALOG(AB)
C       WRITE(6,*)'  Y = ',Y,'   Z = ',Z0,'   FZ = ',FZ0,'   AREA = ',
C     * AREA
C
        DO 10 I=1,2000
        Y=Y+YDEL
        Z1=Y+ADL*(Y**(1.0/AB))
        P=Y**(1.0-(1.0/AB))
        ARG=(P+ADL)/(P+RATIO)
        T1=ALOG(ARG)
        T2=EXP(-Z1)
        FZ1=T1*T2
        AREA=AREA+(FZ0+FZ1)*(Z1-Z0)/2.0
C       WRITE(6,*)'  Y = ',Y,'   Z = ',Z1,'   FZ = ',FZ1,'   AREA = ',
C     * AREA
        FZ0=FZ1
        Z0=Z1
 10     CONTINUE
C
        AREA0=AREA
        YDEL=0.01
C
 15     Y=Y+YDEL
        Z1=Y+ADL*(Y**(1.0/AB))
        P=Y**(1.0-(1.0/AB))
        ARG=(P+ADL)/(P+RATIO)
        T1=ALOG(ARG)
        T2=EXP(-Z1)
        FZ1=T1*T2
        AREA=AREA+(FZ0+FZ1)*(Z1-Z0)/2.0
        IF(ABS(AREA-AREA0).LE.1.0E-08)GO TO 20
        AREA0=AREA
C       WRITE(6,*)'  Y = ',Y,'   Z = ',Z1,'   FZ = ',FZ1,'   AREA = ',
C     * AREA
        FZ0=FZ1
        Z0=Z1
```

```
      GO TO 15
C
  20  RETURN
      END
C
C
      SUBROUTINE SIGMA(AB,ADL,DSUM)
      DOUBLE PRECISION FAC,ARG,XJ,DAB,DADL
      DOUBLE PRECISION SIGN,G,FF,FF1,T1,T2,DSUM,G1,RT
      SIGN=-1.0D0
      FAC=1.0D0
      T1=0.0D0
      T2=0.0D0
      DAB=AB
      DADL=ADL
      J=1
      G1=1.0D0
 10   XJ=DFLOAT(J)
      ARG=XJ/DAB
C     IF(ARG.GE.57.0D0)GO TO 20
      G=DGAMMA(ARG)
C     WRITE(6,*)'  J = ',J,'   ARG = ',ARG,' G = ',G
      RT=G/G1
      FAC=(DADL/XJ)*(FAC*RT)*SIGN
      FF=FAC
C     FF1=FF*XJ
      IF(DABS(FF).LE.1.0D-12)GO TO 20
      G1=RT*G1
      T1=T1+FF
C     T2=T2+FF1
      J=J+1
      GO TO 10
 20   DSUM=T1
C20   DSUM=T2
      RETURN
      END
$ENTRY
0.01 0.05 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.9
//GO.FT09F001 DD DSN=CEFIOR.SAS.DATA3,DISP=SHR
//GO.FT10F001 DD DSN=CEFIOR.SAS.DATA4,DISP=SHR
$$
//
```

```
//PROJECT JOB (1304,57931,1,20),'ARORA',MSGCLASS=S,CLASS=A
/*ROUTE PRINT CEBA
/*JOBPARM SHIFT=N
//      EXEC WATFIV,REGION.GO=4000K,TIME.GO=200
$JOB            TIME=60
C=============================================================
C            TWO COMPONENT EXTREME VALUE DISTRIBUTION (TCEV)
C     ---------------------------------------------------------
C       THIS PROGRAM COMPUTES THE NUMERICAL VALUES OF THE APPROXIMATION
C       SUGGESTED FOR CONSTRAINT C4, FOR VARIOUS ADL AND AB COMBINATIONS
C=============================================================
        REAL LNADL
C
        DEL=0.5
        AB=1.5
C
        T1=0.1*EXP(-1.)
        DO 10 I=1,14
        WRITE(6,*)'        '
        WRITE(6,*)'  AB = ',AB
        WRITE(6,*)'  --------------------------'
        WRITE(6,*)'        '
        LNADL=-5.0
        ADL=EXP(LNADL)
        DO 20 J=1,11
        T2=(3.+AB)**2.059
C
        POWER=ALOG(3.0)-2.*(5.5**(-AB))
        T3=ADL**POWER
C
        C4=T1*T2*T3
        WRITE(6,*)'  LNADL = ',LNADL,'  ADL = ',ADL,'  C4 = ',C4
        WRITE(6,*)'        '
        LNADL=LNADL+DEL
        ADL=EXP(LNADL)
  20    CONTINUE
        AB=AB+DEL
  10    CONTINUE
        STOP
        END
$ENTRY
$$
//
C
```

```
//PROJECT JOB (1304,77493,1,20),'ARORA',MSGCLASS=S,CLASS=B
/*ROUTE PRINT CEBA
//     EXEC FORTVCLG,REGION.GO=4000K,TIME.GO=200
/*JOBPARM SHIFT=D
//FORT.SYSIN DD *
C
      DIMENSION FPR(4),QNTL(4)
      DATA FPR(1),FPR(2),FPR(3),FPR(4)/0.5,0.9,0.99,0.999/
C=====================================================================
C                TWO COMPONENT EXTREME VALUE DISTRIBUTION
C---------------------------------------------------------------------
C           CALCULATION OF QUANTILES FROM POPULATION PARAMETERS
C=====================================================================
C
      AB1=10.00
      ADL1=10.00
      AB2=3.067*AB1
      ADL2=0.173*(ADL1**(1.0/3.067))
      DO 12 I=1,4
      FF=FPR(I)
      CALL TCEVVA(ADL1,ADL2,AB1,AB2,FF,XX)
      QNTL(I)=XX
  12  CONTINUE
          WRITE(6,*)(QNTL(J),J=1,4)
C
      STOP
      END
C
      SUBROUTINE TCEVVA(ADL1,ADL2,AB1,AB2,FF,XX)
C=====================================================
C  SUBROUTINE PER IL CALCOLO DI X UNA VOLTA ASSEGNATO
C  UN VALORE DELLA F(X) DEL MODELLO TCEV
C=====================================================
      REAL*8 DL1,DL2,B1,B2,F,X,FX
      COMMON DL1,DL2,B1,B2
      DL1=ADL1
      DL2=ADL2
      B1=AB1
      B2=AB2
      F=FF
      X=B1*DLOG(DL1)-B1*ALOG(ALOG(1/.35))
10    FR=FX(X)
      IF(ABS(FR-F).LT.0.0000001) THEN
      GOTO 57
      ELSE
      X=X-(FR-F)/F1X(X)
      END IF
      GOTO 10
 57   CONTINUE
      XX=X
      RETURN
      END
C
      FUNCTION FX(X)
```

```
      REAL*8 FX
      REAL*8 DL1,DL2,B1,B2,X
      COMMON DL1,DL2,B1,B2
      FX=DEXP(-DL1*DEXP(-X/B1)-DL2*DEXP(-X/B2))
C     WRITE(6,*)FX
      RETURN
      END
C
      REAL FUNCTION F1X(X)
      REAL*8 DL1,DL2,B1,B2,X,FX
      COMMON DL1,DL2,B1,B2
      F1X=FX(X)*((DL1/B1)*DEXP(-X/B1)+(DL2/B2)*DEXP(-X/B2))
      RETURN
      END
//GO.SYSIN DD *
//
//*GO.FT08F001 DD DSN=CEAROR.OUT.G21,DISP=SHR
//*GO.FT09F001 DD DSN=CEAROR.OUT.G22,DISP=SHR
C--------   ITERATIVE ESTIMATION OF AB1
C           USING IMSL ROUTINE ZREAL2  ------------
C
C     EPS=1.0E-5
C     EPS2=1.0E-5
C     ETA=1.0E-2
C     NSIG=4
C     ITMAX=100
C     N1=1
C     THETA1=10.0
C     CALL ZREAL2(FF,EPS,EPS2,ETA,NSIG,N1,THETA1,ITMAX,IER)
C     STHETA1=STHETA1+THETA1
C     WRITE(6,*)THETA1,ITMAX
C
```

```
//CEFIOR2 JOB (1304,77493,,20),'ARORA',MSGCLASS=S,CLASS=Q,
//       NOTIFY=CEFIOR2
// EXEC SAS,REGION=2048K
/*JOBPARM SHIFT=D
//ONE DD DSN=CEFIOR.SAS.DATA1,DISP=SHR
//TWO DD DSN=CEFIOR.SAS.DATA2,DISP=SHR
          GOPTIONS DEVICE=BEN9215 HSIZE=8 VSIZE=8;

          DATA CURVE1;INFILE ONE;
          INPUT X1 Y1 Y2 Y3 Y4 Y5 Y6 Y7;
          INFILE TWO;
          INPUT X2 Y8 Y9 Y10 Y11 Y12 Y13 Y14;
          PROC GPLOT DATA=CURVE1;
          SYMBOL1 V=PLUS I=SPLINE L=1;
          SYMBOL2 V=PLUS I=SPLINE L=1;
          SYMBOL3 V=PLUS I=SPLINE  L=1;
          SYMBOL4 V=PLUS I=SPLINE  L=1;
          SYMBOL5 V=PLUS I=SPLINE L=1;
          SYMBOL6 V=PLUS I=SPLINE L=1;
          SYMBOL7 V=PLUS I=SPLINE L=1;
          SYMBOL8 V=PLUS I=SPLINE  L=1;
          SYMBOL9 V=PLUS I=SPLINE  L=1;
          SYMBOL10 V=PLUS I=SPLINE L=1;
          SYMBOL11 V=PLUS I=SPLINE L=1;
          SYMBOL12 V=PLUS I=SPLINE L=1;
          SYMBOL13 V=PLUS I=SPLINE  L=1;
          SYMBOL14 V=PLUS I=SPLINE  L=1;
          TITLE1 'F1 VERSUS -LN(ADL) FOR SEVERAL AB VALUES';
          LABEL   X1='-LN(ADL)';
          LABEL   Y1=' F1';
          PLOT Y1*X1 Y2*X1 Y3*X1 Y4*X1 Y5*X1 Y6*X1 Y7*X1
               Y8*X2 Y9*X2 Y10*X2 Y11*X2 Y12*X2 Y13*X2 Y14*X2/
                        HAXIS= 0 TO 5 BY 0.5
                        VAXIS= 0 TO 5 BY 0.5;
//


//*        DATA CURVE2;INFILE TWO;
//*        INPUT XG YG;
//*        PROC GPLOT;
//*        PLOT YG*XG=1/OVERLAY HREF=0 VREF=0
//*                        HAXIS= -3 TO 3 BY 1 VAXIS= -3 TO 3 BY
//*        SYMBOL1  V=STAR;
//*        TITLE1 .H=3 .F=DUPLEX GOAL POINTS HISTORY;
//*        TITLE2 .H=1 .F=ITALIC FOR 5 SESSIONS;
//*        TITLE3 .H=1 .F=ITALIC (K=4  Q=4);
//*        LABEL YG=Y;
//*        LABEL XG=X;
//
          FOOTNOTE1  SQUARES....( T = 1500 SEC);
          FOOTNOTE2  PLUS.......( T = 1600 SEC);
          FOOTNOTE3  TRIANGLE...( T = 1800 SEC);
          FOOTNOTE4  HASH.......( T = 2000 SEC);
          FOOTNOTE5  STAR.......( T = 2200 SEC);
```