```
C REMARK ON ALGORITHM 500
      PROGRAM MAIN
C
C MINIMIZATION OF UNCONSTRAINED MULTIVARIATE FUNCTIONS
C
C BY D.F. SHANNO AND K.H. PHUA
C
C ACM TRANSACTIONS ON MATHEMATICAL SOFTWARE 6 (DECEMBER 1980),
C 618-622
C
C TEST PROGRAM TO TEST CONMIN ON WOODS FUNCTION                          CNMN0001
C                                                                       CNMN0002
      DOUBLE PRECISION X(4),G(4),W(22),EPS,F,ACC                        CNMN0003
C                                                                       CNMN0004
C SET STOPPING CRITERIA                                                 CNMN0005
C                                                                       CNMN0006
      ACC=10.D-20                                                       CNMN0007
      EPS=.001                                                          CNMN0008
C                                                                       CNMN0009
C SET METHOD TO CONJUGATE GRADIENT METHOD                               CNMN0010
C                                                                       CNMN0011
      NMETH=1                                                           CNMN0012
C                                                                       CNMN0013
C SET INITIAL ESTIMATES TO THE MINIMIZER                                CNMN0014
C                                                                       CNMN0015
10    X(1)=-2.0                                                         CNMN0016
      X(2)=-1.0                                                         CNMN0017
      X(3)=-2.0                                                         CNMN0018
      X(4)=-1.0                                                         CNMN0019
C*************************************************************************CNMN0020
C THIS STATEMENT MUST BE CHANGED TO SET THE SYSTEM OUTPUT                CNMN0021
      IDEV=6                                                            CNMN0022
C*************************************************************************CNMN0023
      CALL CONMIN(4,X,F,G,IFUN,ITER,EPS,NFLAG,300,W,1,22,IDEV,ACC,NMETH)CNMN0024
C                                                                       CNMN0025
C TEST FOR CONVERGENCE                                                  CNMN0026
C                                                                       CNMN0027
      IF(NFLAG.GT.0)GO TO 20                                            CNMN0028
C                                                                       CNMN0029
C WRITE NORMAL OUTPUT                                                   CNMN0030
C                                                                       CNMN0031
      WRITE(IDEV,100)F,ITER,IFUN                                        CNMN0032
      GO TO 30                                                          CNMN0033
C                                                                       CNMN0034
C WRITE ERROR OUTPUT                                                    CNMN0035
C                                                                       CNMN0036
20    WRITE(IDEV,130)F,ITER,IFUN,NFLAG                                  CNMN0037
C                                                                       CNMN0038
C WRITE FINAL X AND G VECTORS                                           CNMN0039
C                                                                       CNMN0040
30    WRITE(IDEV,110)(X(I),I=1,4)                                       CNMN0041
      WRITE(IDEV,120)(G(I),I=1,4)                                       CNMN0042
C                                                                       CNMN0043
C TEST IF THE QUASI-NEWTON METHOD HAS BEEN TESTED                       CNMN0044
C                                                                       CNMN0045
      IF(NMETH.EQ.1)GO TO 40                                            CNMN0046
C                                                                       CNMN0047
```

```
C TEST QUASI-NEWTON METHOD                                              CNMN0048
C                                                                       CNMN0049
       NMETH=1                                                          CNMN0050
C       NMETH=0
C      GO TO 10
CNMN0051
40     STOP                                                             CNMN0052
100    FORMAT(13H0CONVERGENCE./11H FINAL F = ,D16.8,12H ITERATIONS ,I3,  CNMN0053
     116H FUNCTION CALLS ,I4)                                           CNMN0054
110    FORMAT(9H0FINAL X./1H ,4D16.8)                                   CNMN0055
120    FORMAT(9H0FINAL G./1H ,4D16.8)                                   CNMN0056
130    FORMAT(11H0FINAL F = ,D16.8,8H ITER = ,I3,8H IFUN = ,I4,         CNMN0057
     19H NFLAG = ,I1)                                                   CNMN0058
       END                                                              CNMN0059
C                                                                       CNMN0060
C SUBROUTINE TO CALCULATE F AND G                                       CNMN0061
C F CONTAINS THE FUNCTION VALUE                                         CNMN0062
C G(1),...,G(4) CONTAIN THE GRADIENT ELEMENTS                           CNMN0063
C N AND X(1),...,X(4) ARE INPUT TO THE SUBROUTINE                       CNMN0064
C                                                                       CNMN0065
       SUBROUTINE CALCFG(N,X,F,G)                                       CNMN0066
       DOUBLE PRECISION X(N),G(N),F,A,B
       A=X(2)-X(1)*X(1)
       B=X(4)-X(3)*X(3)
       F=100.*A*A+(1.-X(1))**2+90.*B*B+(1.-X(3))**2
     1+10.1*((X(2)-1.)**2+(X(4)-1.)**2)+19.8*(X(2)-1.)*(X(4)-1.)
       G(1)=-2.*(200.*X(1)*A+1.-X(1))
       G(2)=2.*(100.*A+10.1*(X(2)-1.)+9.9*(X(4)-1.))
       G(3)=-2.*(180.*X(3)*B+1.-X(3))
       G(4)=2.*(90.*B+10.1*(X(4)-1.)+9.9*(X(2)-1.))
       RETURN
       END
       SUBROUTINE CONMIN(N,X,F,G,IFUN,ITER,EPS,NFLAG,MXFUN,W,           CNMN0078
     1IOUT,MDIM,IDEV,ACC,NMETH)
C
C PURPOSE:    SUBROUTINE CONMIN MINIMIZES AN UNCONSTRAINED NONLINEAR
C             SCALAR VALUED FUNCTION OF A VECTOR VARIABLE X
C             EITHER BY THE BFGS VARIABLE METRIC ALGORITHM OR BY A
C             BEALE RESTARTED CONJUGATE GRADIENT ALGORITHM.
C
C USAGE:      CALL CONMIN(N,X,F,G,IFUN,ITER,EPS,NFLAG,MXFUN,W,
C             IOUT,MDIM,IDEV,ACC,NMETH)
C
C PARAMETERS: N      THE NUMBER OF VARIABLES IN THE FUNCTION TO
C                    BE MINIMIZED.
C             X      THE VECTOR CONTAINING THE CURRENT ESTIMATE TO
C                    THE MINIMIZER. ON ENTRY TO CONMIN,X MUST CONTAIN
C                    AN INITIAL ESTIMATE SUPPLIED BY THE USER.
C                    ON EXITING,X WILL HOLD THE BEST ESTIMATE TO THE
C                    MINIMIZER OBTAINED BY CONMIN. X MUST BE DOUBLE
C                    PRECISIONED AND DIMENSIONED N.
C             F      ON EXITING FROM CONMIN,F WILL CONTAIN THE LOWEST
C                    VALUE OF THE OBJECT FUNCTION OBTAINED.
C                    F IS DOUBLE PRECISIONED.
C             G      ON EXITING FROM CONMIN,G WILL CONTAIN THE
C                    ELEMENTS OF THE GRADIENT OF F EVALUATED AT THE
C                    POINT CONTAINED IN X. G MUST BE DOUBLE
```

```
C                   PRECISIONED AND DIMENSIONED N.
C              IFUN   UPON EXITING FROM CONMIN,IFUN CONTAINS THE
C                     NUMBER OF TIMES THE FUNCTION AND GRADIENT
C                     HAVE BEEN EVALUATED.
C              ITER   UPON EXITING FROM CONMIN,ITER CONTAINS THE
C                     TOTAL NUMBER OF SEARCH DIRECTIONS CALCULATED
C                     TO OBTAIN THE CURRENT ESTIMATE TO THE MINIZER.
C              EPS    EPS IS THE USER SUPPLIED CONVERGENCE PARAMETER.
C                     CONVERGENCE OCCURS WHEN THE NORM OF THE GRADIENT
C                     IS LESS THAN OR EQUAL TO EPS TIMES THE MAXIMUM
C                     OF ONE AND THE NORM OF THE VECTOR X. EPS
C                     MUST BE DOUBLE PRECISIONED.
C              NFLAG  UPON EXITING FROM CONMIN,NFLAG STATES WHICH
C                     CONDITION CAUSED THE EXIT.
C                     IF NFLAG=0, THE ALGORITHM HAS CONVERGED.
C                     IF NFLAG=1, THE MAXIMUM NUMBER OF FUNCTION
C                        EVALUATIONS HAVE BEEN USED.
C                     IF NFLAG=2, THE LINEAR SEARCH HAS FAILED TO
C                        IMPROVE THE FUNCTION VALUE. THIS IS THE
C                        USUAL EXIT IF EITHER THE FUNCTION OR THE
C                        GRADIENT IS INCORRECTLY CODED.
C                     IF NFLAG=3, THE SEARCH VECTOR WAS NOT
C                        A DESCENT DIRECTION. THIS CAN ONLY BE CAUSED
C                        BY ROUNDOFF,AND MAY SUGGEST THAT THE
C                        CONVERGENCE CRITERION IS TOO STRICT.
C              MXFUN  MXFUN IS THE USER SUPPLIED MAXIMUM NUMBER OF
C                     FUNCTION AND GRADIENT CALLS THAT CONMIN WILL
C                     BE ALLOWED TO MAKE.
C              W      W IS A VECTOR OF WORKING STORAGE.IF NMETH=0,
C                     W MUST BE DIMENSIONED 5*N+2. IF NMETH=1,
C                     W MUST BE DIMENSIONED N*(N+7)/2. IN BOTH CASES,
C                     W MUST BE DOUBLE PRECISIONED.
C              IOUT   IOUT IS A USER  SUPPLIED OUTPUT PARAMETER.
C                     IF IOUT = 0, THERE IS NO PRINTED OUTPUT FROM
C                     CONMIN. IF IOUT > 0,THE VALUE OF F AND THE
C                     NORM OF THE GRADIENT SQUARED,AS WELL AS ITER
C                     AND IFUN,ARE WRITTEN EVERY IOUT ITERATIONS.
C              MDIM   MDIM IS THE USER SUPPLIED DIMENSION OF THE
C                     VECTOR W. IF NMETH=0,MDIM=5*N+2. IF NMETH=1,
C                     MDIM=N*(N+7)/2.
C              IDEV   IDEV IS THE USER SUPPLIED NUMBER OF THE OUTPUT
C                     DEVICE ON WHICH OUTPUT IS TO BE WRITTEN WHEN
C                     IOUT>0.
C              ACC    ACC IS A USER SUPPLIED ESTIMATE OF MACHINE
C                     ACCURACY. A LINEAR SEARCH IS UNSUCCESSFULLY
C                     TERMINATED WHEN THE NORM OF THE STEP SIZE
C                     BECOMES SMALLER THAN ACC. IN PRACTICE,
C                     ACC=10.D-20 HAS PROVED SATISFACTORY. ACC IS
C                     DOUBLE PRECISIONED.
C              NMETH  NMETH IS THE USER SUPPLIED VARIABLE WHICH
C                     CHOOSES THE METHOD OF OPTIMIZATION. IF
C                     NMETH=0,A CONJUGATE GRADIENT METHOD IS
C                     USED. IF NMETH=1, THE BFGS METHOD IS USED.
C
C REMARKS:    IN ADDITION TO THE SPECIFIED VALUES IN THE ABOVE
C             ARGUMENT LIST, THE USER MUST SUPPLY A SUBROUTINE
C             CALCFG WHICH CALCULATES THE FUNCTION AND GRADIENT AT
```

```
C             X AND PLACES THEM IN F AND G(1),...,G(N) RESPECTIVELY.
C             THE SUBROUTINE MUST HAVE THE FORM:
C                    SUBROUTINE CALCFG(N,X,F,G)
C                    DOUBLE PRECISION X(N),G(N),F
C
C             AN EXAMPLE SUBROUTINE FOR THE ROSENBROCK FUNCTION IS:
C
C                    SUBROUTINE CALCFG(N,X,F,G)
C                    DOUBLE PRECISION X(N),G(N),F,T1,T2
C                    T1=X(2)-X(1)*X(1)
C                    T2=1.0-X(1)
C                    F=100.0*T1*T1+T2*T2
C                    G(1)=-400.0*T1*X(1)-2.0*T2
C                    G(2)=200.0*T1
C                    RETURN
C                    END
C
      DOUBLE PRECISION X(N),G(N),W(MDIM)
      DOUBLE PRECISION F,FP,FMIN,ALPHA,AT,AP,GSQ,DG,DG1
      DOUBLE PRECISION DP,STEP,ACC,DAL,U1,U2,U3,U4,EPS
      DOUBLE PRECISION XSQ,RTST,DSQRT,DMIN1,DMAX1,DABS
      LOGICAL RSW
C
C INITIALIZE ITER,IFUN,NFLAG,AND IOUTK,WHICH COUNTS OUTPUT ITERATIONS.
C
      ITER=0
      IFUN=0
      IOUTK=0
      NFLAG=0
C
C SET PARAMETERS TO EXTRACT VECTORS FROM W.
C W(I) HOLDS THE SEARCH VECTOR,W(NX+I) HOLDS THE BEST CURRENT
C ESTIMATE TO THE MINIMIZER,AND W(NG+I) HOLDS THE GRADIENT
C AT THE BEST CURRENT ESTIMATE.
C
      NX=N
      NG=NX+N
C
C TEST WHICH METHOD IS BEING USED.
C IF NMETH=0, W(NRY+I) HOLDS THE RESTART Y VECTOR AND
C W(NRD+I) HOLDS THE RESTART SEARCH VECTOR.
C
      IF(NMETH.EQ.1)GO TO 10
      NRY=NG+N
      NRD=NRY+N
      NCONS=5*N
      NCONS1=NCONS+1
      NCONS2=NCONS+2
      GO TO 20
C
C IF NMETH=1,W(NCONS+I) HOLDS THE APPROXIMATE INVERSE HESSIAN.
C
10    NCONS=3*N
C
C  CALCULATETHE FUNCTION AND GRADIENT AT THE INITIAL
C POINT AND INITIALIZE NRST,WHICH IS USED TO DETERMINE
C WHETHER A BEALE RESTART IS BEING DONE. NRST=N MEANS THAT THIS
```

```
C ITERATION IS A RESTART ITERATION. INITIALIZE RSW,WHICH INDICATES
C THAT THE CURRENT SEARCH DIRECTION IS A GRADIENT DIRECTION.
C
20    CALL CALCFG(N,X,F,G)
      IFUN=IFUN+1
      NRST=N
      RSW=.TRUE.
C
C CALCULATE THE INITIAL SEARCH DIRECTION , THE NORM OF X SQUARED,
C AND THE NORM OF G SQUARED. DG1 IS THE CURRENT DIRECTIONAL
C DERIVATIVE,WHILE XSQ AND GSQ ARE THE SQUARED NORMS.
C
      DG1=0.
      XSQ=0.
      DO 30 I=1,N
        W(I)=-G(I)
        XSQ=XSQ+X(I)*X(I)
30    DG1=DG1-G(I)*G(I)
      GSQ=-DG1
C
C TEST IF THE INITIAL POINT IS THE MINIMIZER.
C
      IF(GSQ.LE.EPS*EPS*DMAX1(1.0D0,XSQ))RETURN
C
C BEGIN THE MAJOR ITERATION LOOP. NCALLS IS USED TO GUARANTEE THAT
C AT LEAST TWO POINTS HAVE BEEN TRIED WHEN NMETH=0. FMIN IS THE
C CURRENT FUNCTION VALUE.
C
40    FMIN=F
      NCALLS=IFUN
C
C IF OUTPUT IS DESIRED,TEST IF THIS IS THE CORRECT ITERATION
C AND IF SO, WRITE OUTPUT.
C
      IF(IOUT.EQ.0)GO TO 60
      IF(IOUTK.NE.0)GO TO 50
      WRITE(IDEV,500)ITER,IFUN,FMIN,GSQ
50    IOUTK=IOUTK+1
      IF(IOUTK.EQ.IOUT)IOUTK=0
C
C BEGIN LINEAR SEARCH. ALPHA IS THE STEPLENGTH.
C SET ALPHA TO THE NONRESTART CONJUGATE GRADIENT ALPHA.
C
60    ALPHA=ALPHA*DG/DG1
C
C IF NMETH=1 OR A RESTART HAS BEEN PERFORMED, SET ALPHA=1.0.
C
      IF(NRST.EQ.1.OR.NMETH.EQ.1)ALPHA=1.0
C
C IF A GRADIENT DIRECTION IS USED, SET ALPHA=1.0/DSQRT(GSQ),
C WHICH SCALES THE INITIAL SEARCH VECTOR TO UNITY.
C
      IF(RSW)ALPHA=1.0/DSQRT(GSQ)
C
C THE LINEAR SEARCH FITS A CUBIC TO F AND DAL, THE FUNCTION AND ITS
C DERIVATIVE AT ALPHA, AND TO FP AND DP,THE FUNCTION
C AND DERIVATIVE AT THE PREVIOUS TRIAL POINT AP.
```

```
C INITIALIZE AP ,FP,AND DP.
C
      AP=0.
      FP=FMIN
      DP=DG1
C
C SAVE THE CURRENT DERIVATIVE TO SCALE THE NEXT SEARCH VECTOR.
C
      DG=DG1
C
C UPDATE THE ITERATION.
C
      ITER=ITER+1
C
C CALCULATE THE CURRENT STEPLENGTH  AND STORE THE CURRENT X AND G.
C
      STEP=0.
      DO 70 I=1,N
        STEP=STEP+W(I)*W(I)
        NXPI=NX+I
        NGPI=NG+I
        W(NXPI)=X(I)
70    W(NGPI)=G(I)
      STEP=DSQRT(STEP)
C
C BEGIN THE LINEAR SEARCH ITERATION.
C TEST FOR FAILURE OF THE LINEAR SEARCH.
C
80    IF(ALPHA*STEP.GT.ACC)GO TO 90
C
C TEST IF DIRECTION IS A GRADIENT DIRECTION.
C
      IF(.NOT.RSW)GO TO 20
      NFLAG=2
      RETURN
C
C CALCULATE THE TRIAL POINT.
C
90    DO 100 I=1,N
        NXPI=NX+I
100   X(I)=W(NXPI)+ALPHA*W(I)
C
C EVALUATE THE FUNCTION AT THE TRIAL POINT.
C
      CALL CALCFG(N,X,F,G)
C
C TEST IF THE MAXIMUM NUMBER OF FUNCTION CALLS HAVE BEEN USED.
C
      IFUN=IFUN+1
      IF(IFUN.LE.MXFUN)GO TO 110
      NFLAG=1
      RETURN
C
C COMPUTE THE DERIVATIVE OF F AT ALPHA.
C
110   DAL=0.0
      DO 120 I=1,N
```

```
120   DAL=DAL+G(I)*W(I)
C
C TEST WHETHER THE NEW POINT HAS A NEGATIVE SLOPE BUT A HIGHER
C FUNCTION VALUE THAN ALPHA=0. IF THIS IS THE CASE,THE SEARCH
C HAS PASSED THROUGH A LOCAL MAX AND IS HEADING FOR A DISTANT LOCAL
C MINIMUM.
C
      IF(F.GT.FMIN.AND.DAL.LT.0.)GO TO 160
C
C IF NOT, TEST WHETHER THE STEPLENGTH CRITERIA HAVE BEEN MET.
C
      IF(F.GT.(FMIN+.0001*ALPHA*DG).OR.DABS(DAL/DG)
     1.GT.(.9))GO TO 130
C
C IF THEY HAVE BEEN MET, TEST IF TWO POINTS HAVE BEEN TRIED
C IF NMETH=0 AND IF THE TRUE LINE MINIMUM HAS NOT BEEN FOUND.
C
      IF((IFUN-NCALLS).LE.1.AND.DABS(DAL/DG).GT.EPS.AND.
     1NMETH.EQ.0)GO TO 130
      GO TO 170
C
C A NEW POINT MUST BE TRIED. USE CUBIC INTERPOLATION TO FIND
C THE TRIAL POINT AT.
C
130   U1=DP+DAL-3.0*(FP-F)/(AP-ALPHA)
      U2=U1*U1-DP*DAL
      IF(U2.LT.0.)U2=0.
      U2=DSQRT(U2)
      AT=ALPHA-(ALPHA-AP)*(DAL+U2-U1)/(DAL-DP+2.*U2)
C
C TEST WHETHER THE LINE MINIMUM HAS BEEN BRACKETED.
C
      IF((DAL/DP).GT.0.)GO TO 140
C
C THE MINIMUM HAS BEEN BRACKETED. TEST WHETHER THE TRIAL POINT LIES
C SUFFICIENTLY WITHIN THE BRACKETED INTERVAL.
C IF IT DOES NOT, CHOOSE AT AS THE MIDPOINT OF THE INTERVAL.
C
      IF(AT.LT.(1.01*DMIN1(ALPHA,AP)).OR.AT.GT.(.99*DMAX1
     1(ALPHA,AP)))AT=(ALPHA+AP)/2.0
      GO TO 150
C
C THE MINIMUM HAS NOT BEEN BRACKETED. TEST IF BOTH POINTS ARE
C GREATER THAN THE MINIMUM AND THE TRIAL POINT IS SUFFICIENTLY
C SMALLER THAN EITHER.
C
140   IF(DAL .GT.0.0.AND.0.0.LT.AT.AND.AT.LT.(.99*DMIN1(AP,ALPHA)))
     1GO TO 150
C
C TEST IF BOTH POINTS ARE LESS THAN THE MINIMUM AND THE TRIAL POINT
C IS SUFFICIENTLY LARGE.
C
      IF(DAL.LE.0.0.AND.AT.GT.(1.01*DMAX1(AP,ALPHA)))GO TO 150
C
C IF THE TRIAL POINT IS TOO SMALL,DOUBLE THE LARGEST PRIOR POINT.
C
      IF(DAL.LE.0.)AT=2.0*DMAX1(AP,ALPHA)
```

```
C
C IF THE TRIAL POINT IS TOO LARGE, HALVE THE SMALLEST PRIOR POINT.
C
      IF(DAL.GT.0.)AT=DMIN1(AP,ALPHA)/2.0
C
C SET AP=ALPHA, ALPHA=AT,AND CONTINUE SEARCH.
C
150   AP=ALPHA
      FP=F
      DP=DAL
      ALPHA=AT
      GO TO 80
C
C A RELATIVE MAX HAS BEEN PASSED.REDUCE ALPHA AND RESTART THE SEARCH.
C
160   ALPHA=ALPHA/3.
      AP=0.
      FP=FMIN
      DP=DG
      GO TO 80
C
C THE LINE SEARCH HAS CONVERGED. TEST FOR CONVERGENCE OF THE ALGORITHM.
C
170   GSQ=0.0
      XSQ=0.0
      DO 180 I=1,N
        GSQ=GSQ+G(I)*G(I)
180   XSQ=XSQ+X(I)*X(I)
      IF(GSQ.LE.EPS*EPS*DMAX1(1.0D0,XSQ))RETURN
C
C SEARCH CONTINUES. SET W(I)=ALPHA*W(I),THE FULL STEP VECTOR.
C
      DO 190 I=1,N
190   W(I)=ALPHA*W(I)
C
C COMPUTE THE NEW SEARCH VECTOR. FIRST TEST WHETHER A
C CONJUGATE GRADIENT OR A VARIABLE METRIC VECTOR IS USED.
C
      IF(NMETH.EQ.1)GO TO 330
C
C CONJUGATE GRADIENT UPDATE SECTION.
C TEST IF A POWELL RESTART IS INDICATED.
C
      RTST=0.
      DO 200 I=1,N
        NGPI=NG+I
200   RTST=RTST+G(I)*W(NGPI)
      IF(DABS(RTST/GSQ).GT.0.2)NRST=N
C
C IF A RESTART IS INDICATED, SAVE THE CURRENT D AND Y
C AS THE BEALE RESTART VECTORS AND SAVE D'Y AND Y'Y
C IN W(NCONS+1) AND W(NCONS+2).
C
      IF(NRST.NE.N)GO TO 220
      W(NCONS+1)=0.
      W(NCONS+2)=0.
      DO 210 I=1,N
```

```
        NRDPI=NRD+I
        NRYPI=NRY+I
        NGPI=NG+I
        W(NRYPI)=G(I)-W(NGPI)
        W(NRDPI)=W(I)
        W(NCONS1)=W(NCONS1)+W(NRYPI)*W(NRYPI)
210   W(NCONS2)=W(NCONS2)+W(I)*W(NRYPI)
C
C CALCULATE  THE RESTART HESSIAN TIMES THE CURRENT GRADIENT.
C
220   U1=0.0
      U2=0.0
      DO 230 I=1,N
        NRDPI=NRD+I
        NRYPI=NRY+I
        U1=U1-W(NRDPI)*G(I)/W(NCONS1)
230   U2=U2+W(NRDPI)*G(I)*2./W(NCONS2)-W(NRYPI)*G(I)/W(NCONS1)
      U3=W(NCONS2)/W(NCONS1)
      DO 240 I=1,N
        NXPI=NX+I
        NRDPI=NRD+I
        NRYPI=NRY+I
240   W(NXPI)=-U3*G(I)-U1*W(NRYPI)-U2*W(NRDPI)
C
C IF THIS IS A RESTART ITERATION,W(NX+I) CONTAINS THE NEW SEARCH
C VECTOR.
C
      IF(NRST.EQ.N)GO TO 300
C
C NOT A RESTART ITERATION. CALCULATE THE RESTART HESSIAN
C TIMES THE CURRENT Y.
C
250   U1=0.
      U2=0.
      U3=0.
      U4=0.
      DO 260 I=1,N
        NGPI=NG+I
        NRDPI=NRD+I
        NRYPI=NRY+I
        U1=U1-(G(I)-W(NGPI))*W(NRDPI)/W(NCONS1)
        U2=U2-(G(I)-W(NGPI))*W(NRYPI)/W(NCONS1)
     1  +2.0*W(NRDPI)*(G(I)-W(NGPI))/W(NCONS2)
260   U3=U3+W(I)*(G(I)-W(NGPI))
      STEP=0.
      DO 270 I=1,N
        NGPI=NG+I
        NRDPI=NRD+I
        NRYPI=NRY+I
        STEP=(W(NCONS2)/W(NCONS1))*(G(I)-W(NGPI))
     1  +U1*W(NRYPI)+U2*W(NRDPI)
        U4=U4+STEP*(G(I)-W(NGPI))
270   W(NGPI)=STEP
C
C CALCULATE THE DOUBLY UPDATED HESSIAN TIMES THE CURRENT
C GRADIENT TO OBTAIN THE SEARCH VECTOR.
C
```

```
      U1=0.0
      U2=0.0
      DO 280 I=1,N
        U1=U1-W(I)*G(I)/U3
        NGPI=NG+I
280   U2=U2+(1.0+U4/U3)*W(I)*G(I)/U3-W(NGPI)*G(I)/U3
      DO 290 I=1,N
        NGPI=NG+I
        NXPI=NX+I
290   W(NXPI)=W(NXPI)-U1*W(NGPI)-U2*W(I)
C
C CALCULATE THE DERIVATIVE ALONG THE NEW SEARCH VECTOR.
C
300   DG1=0.
      DO 310 I=1,N
        NXPI=NX+I
        W(I)=W(NXPI)
310   DG1=DG1+W(I)*G(I)
C
C IF THE NEW DIRECTION IS NOT A DESCENT DIRECTION,STOP.
C
      IF (DG1.GT.0.)GO TO 320
C
C UPDATE NRST TO ASSURE AT LEAST ONE RESTART EVERY N ITERATIONS.
C
      IF(NRST.EQ.N)NRST=0
      NRST=NRST+1
      RSW=.FALSE.
      GO TO 40
C
C ROUNDOFF HAS PRODUCED A BAD DIRECTION.
C
320   NFLAG=3
      RETURN
C
C A VARIABLE METRIC ALGORITM IS BEING USED. CALCULATE Y AND D'Y.
C
330   U1=0.0
      DO 340 I=1,N
        NGPI=NG+I
        W(NGPI)=G(I)-W(NGPI)
340   U1=U1+W(I)*W(NGPI)
C
C IF RSW=.TRUE.,SET UP THE INITIAL SCALED APPROXIMATE HESSIAN.
C
      IF(.NOT.RSW)GO TO 380
C
C CALCULATE Y'Y.
C
      U2=0.
      DO 350 I=1,N
        NGPI=NG+I
350   U2=U2+W(NGPI)*W(NGPI)
C
C CALCULATE THE INITIAL HESSIAN AS H=(P'Y/Y'Y)*I
C AND THE INITIAL U2=Y'HY AND W(NX+I)=HY.
C
```

```
      IJ=1
      U3=U1/U2
      DO 370 I=1,N
        DO 360 J=I,N
          NCONS1=NCONS+IJ
          W(NCONS1)=0.0
          IF(I.EQ.J)W(NCONS1)=U3
360     IJ=IJ+1
        NXPI=NX+I
        NGPI=NG+I
370   W(NXPI)=U3*W(NGPI)
      U2=U3*U2
      GO TO 430
C
C CALCULATE W(NX+I)=HY AND U2=Y'HY.
C
380   U2=0.0
      DO 420 I=1,N
        U3=0.0
        IJ=I
        IF(I.EQ.1)GO TO 400
        II=I-1
        DO 390 J=1,II
          NGPJ=NG+J
          NCONS1=NCONS+IJ
          U3=U3+W(NCONS1)*W(NGPJ)
390     IJ=IJ+N-J
400     DO 410 J=I,N
          NCONS1=NCONS+IJ
          NGPJ=NG+J
          U3=U3+W(NCONS1)*W(NGPJ)
410     IJ=IJ+1
        NGPI=NG+I
        U2=U2+U3*W(NGPI)
        NXPI=NX+I
420   W(NXPI)=U3
C
C CALCULATE THE UPDATED APPROXIMATE HESSIAN.
C
430   U4=1.0+U2/U1
      DO 440 I=1,N
        NXPI=NX+I
        NGPI=NG+I
440   W(NGPI)=U4*W(I)-W(NXPI)
      IJ=1
      DO 450 I=1,N
        NXPI=NX+I
        U3=W(I)/U1
        U4=W(NXPI)/U1
        DO 450 J=I,N
          NCONS1=NCONS+IJ
          NGPJ=NG+J
          W(NCONS1)=W(NCONS1)+U3*W(NGPJ)-U4*W(J)
450   IJ=IJ+1
C
C CALCULATE THE NEW SEARCH DIRECTION W(I)=-HG AND ITS DERIVATIVE.
C
```

```
      DG1=0.0
      DO 490 I=1,N
        U3=0.0
        IJ=I
        IF(I.EQ.1)GO TO 470
        II=I-1
        DO 460 J=1,II
          NCONS1=NCONS+IJ
          U3=U3-W(NCONS1)*G(J)
460     IJ=IJ+N-J
470     DO 480 J=I,N
          NCONS1=NCONS+IJ
          U3=U3-W(NCONS1)*G(J)
480     IJ=IJ+1
        DG1=DG1+U3*G(I)
490   W(I)=U3
C
C TEST FOR A DOWNHILL DIRECTION.
C
      IF(DG1.GT.0.)GO TO 320
      RSW=.FALSE.
      GO TO 40
500   FORMAT(10H ITERATION,I5,20H      FUNCTION CALLS,I6/5H F = ,
     1D15.8,13H G-SQUARED = ,D15.8/)
      END
```