AD 742-098

LIBRARY
TECHNICAL REPORT SECTION
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA 93940

PAPER P-801

COMPUTER PROGRAM FOR SOLVING MATHEMATICAL PROGRAMS WITH NONLINEAR PROGRAMS IN THE CONSTRAINTS

Jerome Bracken
James T. McGill

March 1972

INSTITUTE FOR DEFENSE ANALYSES
PROGRAM ANALYSIS DIVISION

BEST AVAILABLE COPY

IDA Log No. HQ 71-13240
Copy 55 of 125 copies

The work reported in this publication was conducted under IDA's Independent Research Program. Its publication does not imply endorsement by the Department of Defense or any other Government agency, nor should the contents be construed as reflecting the official position of any Government agency.

PAPER P-801

COMPUTER PROGRAM FOR SOLVING MATHEMATICAL PROGRAMS WITH NONLINEAR PROGRAMS IN THE CONSTRAINTS

Jerome Bracken
James T. McGill

March 1972

INSTITUTE FOR DEFENSE ANALYSES
PROGRAM ANALYSIS DIVISION
400 Army-Navy Drive, Arlington, Virginia 22202

IDA Independent Research Program

ABSTRACT

This paper documents a computer program to be used in solving nonlinear programming problems with nonlinear programming problems in the constraints. The program, named INSUMT, is used with the standard program, named SUMT, which implements the sequential unconstrained minimization technique for nonlinear programming. SUMT calls INSUMT when it is necessary to solve a nonlinear program in a constraint. The INSUMT program, together with a fairly complete example of its use, is included in the documentation.

Theory and applications of the models which can be solved using this program are treated in two companion papers.

CONTENTS

	Page
INTRODUCTION	1
PROBLEM DESCRIPTION AND SUMMARY OF COMPUTATIONAL PROCEDURE	3
DESCRIPTION OF COMPUTER PROGRAM	7
EXAMPLE PROBLEM	11
COMPUTER PROGRAM INCLUDING USER-SUPPLIED SUBROUTINES FOR EXAMPLE PROBLEM	13
User-Supplied Subroutines for Outside Program	15
INSUMT	21
User-Supplied Subroutines for Inside Programs	63
USER-SUPPLIED INFORMATION CARDS AND RESULTS PRINTING OUT ALL POINTS OF INSIDE PROGRAMS	69
User-Supplied Information Cards	71
Printout of Results	73
RESULTS PRINTING OUT ALL POINTS OF OUTSIDE PROGRAM	91
Printout of Results	93
REFERENCES	101

INTRODUCTION

Reference [2] formulates a class of mathematical programs with optimization problems in the constraints. Reference [3] gives models of a number of defense problems which can be approached using the techniques. The present paper documents a computer program for solving mathematical programs with nonlinear programming problems in the constraints.

Fiacco and McCormick in Reference [4] present theory and computational aspects of the sequential unconstrained minimization technique (SUMT) for nonlinear programming. Applications of nonlinear programming to a number of practical problems are presented in Reference [1]. The initial computer program for SUMT is documented in Reference [5]. The most recent computer program for SUMT is documented in Reference [6]. It handles equality constraints, and includes routines for computation of first and second partial derivatives using function values. Various unconstrained minimization methods are available in the SUMT program, requiring function values and first partial derivatives or requiring function values and first and second partial derivatives. Numerical differentiation routines facilitate the use of various unconstrained minimization methods. These are necessary in solving mathematical programs with nonlinear programs in the constraints, since the partial derivatives of the constraints cannot be stated explicitly.

The present paper is designed to be a supplement to Reference [6] and to be used with it.

PROBLEM DESCRIPTION AND SUMMARY OF COMPUTATIONAL PROCEDURE

One of the mathematical programs considered in Reference [2] is to choose vectors $x = (x_1, \ldots, x_n)$ and $v^i = (v_1^i, \ldots, v_{k_i}^i)$ for $i = 1, \ldots, m$ to

$$\underset{x \in X}{\text{minimize}}\ f(x)$$

subject to

$$h_i(x) = \min_{v^i \in V^i} g^i(x, v^i) \geq 0 \quad , \quad i = 1, 2, \ldots, m \quad .$$

It is shown there that if $g^i(x, v^i)$ is concave in x on a convex set X for every $v^i \in V^i$, then $h_i(x)$ is concave on X where the convex set X may be defined by inequality and/or equality constraints. If, in addition, $f(x)$ is a convex function on X, then the mathematical program is convex.

To outline the computational technique, it is useful to differentiate between the "outside program,"

$$\min_{x \in X} f(x)$$

subject to

$$h_i(x) \geq 0 \quad , \quad i = 1, 2, \ldots, m \quad ,$$

and the "ith inside program,"

$$\min_{v^i \in V^i} g^i(x, v^i) \quad .$$

When convenient the distinction among the m inside programs will be dropped, considering only the generic problem:

$$\min_{v \in V} g(x, v) \quad .$$

The constraint functions $h_i(x)$ in the outside problem are implicit in that their values depend upon the solution of the inside problem, which in turn depends upon the value of x. Thus, a solution technique for the overall problem must not rely on an explicit functional form for $h_i(x)$.

The computer program described in this paper, called INSUMT, is based on SUMT. The standard SUMT program is used for the outside problem. The new INSUMT program is used to solve the inside problem. SUMT and INSUMT are iterative routines. Let x^k denote the value of x for iteration k of the outside problem and v^ℓ denote iteration ℓ for the inside problem. The solution procedure is initialized by the user supplying x^o and v^o.

During iteration k for the outside problem, the inside problem is to choose $v \in V$ to minimize $g(x^k, v)$. At iteration ℓ of the inside problem, there is a value v^ℓ. This value is then used by INSUMT to generate a new value $v^{\ell+1}$. Continuing in this manner, the inside problem is solved, yielding $v^*(x^k)$ such that

$$g(x^k, v^*(x^k)) = \min_{v \in V} g(x^k, v) \quad .$$

Control then passes to the outside problem which generates a new solution value x^{k+1}. The process is repeated until the sequence $\{x^k\}$ converges.

The routine INSUMT resides in core with SUMT and may be called to evaluate more than one constraint of the outside problem. For each such constraint, user-supplied subroutines of INSUMT provide information concerning the functional form of $g^i(x,v^i)$ and of the constraint set V^i. User-supplied subroutines for SUMT give analogous information for $f(x)$ and X.

The SUMT and INSUMT programs iteratively generate values of the solution variable by optimizing an unconstrained penalty function. Specifically, iteration $\ell + 1$ generates $v^{\ell+1}$ from v^{ℓ} and from the functional forms of the objective function of the inside program and the constraint set. The computation calls for the first, and sometimes second, partial derivatives of the relevant functions. These may be specified by the user. However, for the outside problem, explicit derivatives of $h_i(x)$ cannot be given. SUMT uses numerical differentiation in this case to approximate the derivatives to be used in generating x^{k+1} from x^k.

DESCRIPTION OF COMPUTER PROGRAM

The program is modular in structure. It consists basically of the SUMT program (subroutines MAIN through XMOVE), one set of user-supplied subroutines containing function evaluations for the outside mathematical program (READIN, RESTNT, GRAD1 and MATRIX), the INSUMT program (subroutines SUB through XMOVES), and one set of user-supplied subroutines containing function evaluations for the one or more inside mathematical programs (READIX, RESTNX, GRAD1S and MATRXX).

The workings of the SUMT subroutines MAIN through XMOVE are described in Reference [6]. The user-supplied subroutines for the outside program perform the following tasks when used with INSUMT.

READIN reads in the parameters used in evaluating the objective function and constraints.

RESTNT evaluates the objective function and constraints of the outside program for a value of x. Except where constraints contain inside mathematical programs, the functions are evaluated for the current value of x. Otherwise, RESTNT calls SUB, identifying the inside mathematical program to be solved for the current value of x, and SUB returns the value $v^*(x)$. RESTNT then evaluates $h(x) = g(x, v^*(x))$.

GRAD1 evaluates the first partial derivatives of the objective function and constraints in two ways. Where the constraints do not contain inside mathematical programs, for the current x the first partial derivatives are evaluated either explicitly or using numerical differentiation. Otherwise, GRAD1 uses DIFF1 to numerically differentiate the constraints which include inside mathematical programs.

MATRIX evaluates the second partial derivatives of the objective function and constraints of the outside mathematical program in two

ways. Where the constraints do not contain inside mathematical programs, for the current x the second partial derivatives are evaluated either explicitly or using numerical differentiation. Otherwise, MATRIX uses DIFF2 to numerically differentiate the constraints which include inside mathematical programs.

The INSUMT program consists of subroutines SUB through XMOVES. Three basic changes are made in SUMT to result in INSUMT.

First, all subroutine names are changed by simply adding an S to the end except where this results in more than six letters, in which case the last two letters are replaced by X (e.g., OPT becomes OPTS and RESTNT becomes RESTNX). All calls to subroutines within INSUMT are changed to include the revised names of the called subroutines. SUMT calls INSUMT only by RESTNT calling SUB. After SUB is called INSUMT calls only the subroutines within INSUMT until SUB returns to RESTNT.

Second, all labeled COMMON arrays are changed by adding an S to the end, except where this results in more than six letters, in which case the last two letters are replaced by X (e.g., SHARE becomes SHARES and CONPAR becomes CONPAX).

Third, subroutine MAIN is modified to obtain subroutine SUB by changing the name from MAIN to SUB, and by modifying the subroutine to read in data for the inside mathematical programs the first time each problem is solved and to save the data in an array for subsequent solutions of the problems.

User-supplied subroutines READIX, RESTNX, GRAD1S and MATRXX are called by INSUMT. Depending on a parameter denoting the inside mathematical program being solved, the subroutines calculate appropriate function values, first partial derivatives, or second partial derivatives.

In the first several runs of a program, it is necessary to observe the intermediate points of the solution of the inside programs. RESTNT thus prints out which inside program is being solved, and INSUMT prints out the intermediate points. After it

is established that the inside programs are being successfully solved, printing of their points is suppressed by deleting the print statements from RESTNT and by modifying BODYS, CONVRX, ESTIMS, FEASS, INVERX, OPTS, OUTPUX, PUNCHS, TECHECX and TIMECS. The modifications are described later.

EXAMPLE PROBLEM

An example problem is used to describe the use of INSUMT with SUMT.

The problem is to choose $x_1, \ldots, x_n$ to

$$\text{minimize } x_1 + \ldots + x_n$$

subject to

$$\left[\underset{v_1 + \ldots + v_n \leq n}{\text{minimum}} \; x_1(v_1 - 2)^2 + \ldots + x_n(v_n - 2)^2 \right] \geq r_1$$

$$\left[\underset{v_1^2 + \ldots + v_n^2 \leq n}{\text{minimum}} \; x_1^{.5}(v_1 - 2)^2 + \ldots + x_n^{.5}(v_n - 2)^2 \right] \geq r_2 \; .$$

To illustrate the problem, take $n = 4$, $r_1 = 4$, $r_2 = 4$ and let the starting point be $x_1 = x_2 = x_3 = x_4 = 4$ for the outside program and $v_1 = v_2 = v_3 = v_4 = .5$ for both inside programs. The value of the objective function of the outside program is

$$x_1 + x_2 + x_3 + x_4 = 16 \quad .$$

The objective function of the first inside program is

$$4(.5 - 2)^2 + 4(.5 - 2)^2 + 4(.5 - 2)^2 + 4(.5 - 2)^2 = 36 \quad ,$$

and since $.5 + .5 + .5 + .5 = 2 \leq 4$ the first inside program starting point is feasible. For the second inside program the objective

function is

$$2(.5 - 2)^2 + 2(.5 - 2)^2 + 2(.5 - 2)^2 + 2(.5 - 2)^2 = 18 \quad ,$$

and since $.5^2 + .5^2 + .5^2 + .5^2 = 1 \leq 4$ the second inside program starting point is feasible. Since $36 > r_1 \ (= 4)$, $18 > r_2 \ (= 4)$, the three starting points, one outside and two inside, provide a feasible point for all three programs.

The solution to the example is $x_1 = x_2 = x_3 = x_4 = 1$, with $v_1 = v_2 = v_3 = v_4 = 1$ in both inside programs, yielding

$$x_1 + x_2 + x_3 + x_4 = 1 + 1 + 1 + 1 = 4$$

and

$$\left[\begin{array}{l} 1(1 - 2)^2 + 1(1 - 2)^2 + 1(1 - 2)^2 + 1(1 - 2)^2 = 4 \\ \text{s.t.} \\ (1 + 1 + 1 + 1 = 4) \leq 4 \end{array}\right] \geq 4$$

$$\left[\begin{array}{l} 1^2(1 - 2)^2 + 1^2(1 - 2)^2 + 1^2(1 - 2)^2 + 1^2(1 - 2)^2 = 4 \\ \text{s.t.} \\ (1^2 + 1^2 + 1^2 + 1^2 = 4) \leq 4 \end{array}\right] \geq 4 \quad .$$

COMPUTER PROGRAM INCLUDING USER-SUPPLIED SUBROUTINES FOR EXAMPLE PROBLEM

In this section the user-supplied subroutines for the outside program are presented, followed by the INSUMT program, followed by the user-supplied subroutines for the inside programs. The SUMT program is not supplied, being documented in Reference [6].

RESTNT contains print statements for each call for solution and end of solution of an inside program. The INSUMT subroutines SUB, BODYS, CHCKEX, CONVRX, ESTIMS, FEASS, INVERX, OPTS, OUTPUX, PUNCHS, TCHECX and TIMECS contain print statements for the points of the inside programs. Thus the program as listed prints all intermediate points. However, the changes necessary to suppress printing are indicated in heavy markings directly on the listing. Simply remove the boxed statements, and replace them by the statements written, if any. Printing is done by SUB only the first time it is called, so SUB is not modified. Printing is done by CHCKEX only if control cards dictate, so it is not modified. All other print statements are removed to suppress printing of inside programs.

It should be noted that all data of the example problem are contained in the set of user-supplied subroutines for the outside program (READIN, RESTNT, GRAD1, and MATRIX) and the set of user-supplied subroutines for the inside programs (READIX, RESTNX, GRAD1S and MATRXX). For most problems READIN should be used to read data for outside and inside programs, since READIX is called each time an inside program is solved, and it should not be used.

Control cards are read in the following order: Outside program, first inside program, second inside program, and so on if there are

more inside programs. SUB saves the control card data, and inside program starting points, and provides these data each time an inside program is solved. Dimensions on PARS are presently (2,47), including up to 20 variables in the inside program. This would have to be changed for more than 2 inside programs or 20 variables.

User-Supplied Subroutines for Outside Program

```
                SUBROUTINE READIN
000002     9999 CONTINUE
000002          RETURN
000003          END
```

```
000005          SUBROUTINE RESTNT(IN,VAL)
000005          COMMON/PROB/ISP
000005          COMMON/OUT/NOUT,XT(100),V(100)
000005          COMMON/SHARE/ X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000005          FN=N
000006          VAL=0.
000007          IF(IN)100,100,200
000010      100 DO 150 J=1,N
000012      150 VAL = VAL + X(J)
000016          GO TO 9999
000016      200 IF(IN-1)300,300,400
000020      300 ISP=1
000021          NOUT=N
000023          DO 305 J=1,N
000024          IF(X(J)-0.)304,304,305
000027      304 X(J)=1.E-08
000031      305 XT(J)=X(J)
000036          WRITE(6,307)
000041      307 FORMAT(11H1 PROBLEM A)
000041          CALL SUB
000042          WRITE(6,308)
000046      308 FORMAT(18H1 PROBLEM A SOLVED)
000046          DO 320 J=1,N
000051      320 VAL = VAL + X(J)*(V(J)-2.)**2
000057          VAL=VAL-4.
000061          GO TO 9999
000061      400 ISP=2
000062          NOUT=N
000064          DO 405 J=1,N
000065          IF(X(J)-0.)404,404,405
000070      404 X(J)=1.E-08
000072      405 XT(J)=X(J)
000076          WRITE(6,407)
000102      407 FORMAT(11H1 PROBLEM B)
000102          CALL SUB
000103          WRITE(6,408)
000107      408 FORMAT(18H1 PROBLEM B SOLVED)
000107          DO 420 J=1,N
000112      420 VAL = VAL + X(J)**.5 * (V(J)-2.)**2
000124          VAL=VAL-4.
000126          GO TO 9999
000126     9999 CONTINUE
000126          RETURN
000127          END
```

```
                SUBROUTINE GRAD1(IN)
000003          COMMON/SHARE/ X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1          10000070
000003          DO 50 J=1,N
000005       50 DEL(J)=0.
000010          IF(IN)100,100,200
000011      100 DO 150 J=1,N
000013      150 DEL(J) = 1.
000017          GO TO 9999
000017      200 CALL DIFF1(IN)
000020          GO TO 9999
000022     9999 RETURN
000023          END
```

```
                SUBROUTINE MATRIX(IN,L)
000005          COMMON/SHARE/ X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1        10000070
000005          IF(IN)100,100,200
000006      100 L=1
000007          GO TO 9999
000010      200 CALL DIFF2(IN)
000011          GO TO 9999
000013     9999 CONTINUE
000013          RETURN
000014          END
```

INSUMT

```
                SUBROUTINE SUB
          C
          C              MARCH 1971
          C
          C MAIN IS THE PROGRAM THAT INITIATES THE SUMT ALGORITHM. THE INPUT OF
          C PARAMETERS, OPTIONS, AND STARTING POINT IS DONE IN MAIN. AFTER THE
          C SOLUTION OF ONE NLP PROBLEM MAIN LOOKS FOR DATA FOR ANOTHER NLP PROB.
          C
000002          COMMON/PROB/ISP
000002          COMMON/OUT/NOUT,XT(100),V(100)
000002          COMMON/IN/W(100)
          C
000002          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NPI,NMI
000002          COMMON /EQALS/H, HI, MZ
000002          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NTIO
000002          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000002          COMMON/CRSTS/DELX(100), DELX0(100), RHOIN,RATIO, EPSI, THETA0,
               1 RSIG1, GI,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJI(200), DOTT, PGRAD(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMINI, NPHASE, NSATIS
000002          COMMON/EXPOPX / NEXOP1, NEXOP2, XEP1, XEP2
          C
000002          DIMENSION IFTS(2),PARS(2,47)
000002          DATA IFTS/0,0/
          C
000002          DO 5 J=1,NOUT
000004        5 W(J)=XT(J)
000010          IF(IFTS(ISP))10,10,20
          C
          C     PARAMETER CARD
000012    10    READ (5,50) EPSI,RHOIN,THETA0,RATIO,TMMAX,M,N,MZ
          C     INITIAL  X VERCTOR CARD  FORMAT
000036          READ (5,60) (X(I),I=1,N)
000051          NTCTR=0
000052          NP1=N+1
000054          NM1=N-1
          C SUBROUTINE READIN IS UNDER PROGRAMMER CONTROL
000055          CALL READIX
          C OPTION CARD FOLLOWS PROGRAMMERS DATA
000056          READ (5,80) NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NTIO
000106          WRITE (6,110)
000112          WRITE (6,120) N,M,MZ,TMMAX,RHOIN,RATIO,EPS1,THETA0
000136          WRITE (6,130)
000142          WRITE (6,80) NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NTIO
          C--READ TOLERANCES
000172          READ (5,60) XEP1, XEP2
000202          WRITE (6,90)
000206          WRITE (6,70) XEP1, XEP2
          C--READ SECOND OPTION CARD
000216          READ (5,80) NEXOP1,NEXOP2
000226          WRITE (6,100)
000232          WRITE (6,80) NEXOP1,NEXOP2
000242          PARS (ISP, 1) = EPS1
000244          PARS(ISP, 2) = RHOIN
000246          PARS (ISP, 3) = THETA0
000247          PARS (ISP, 4) = RATIO
000251          PARS (ISP, 5) = TMMAX
```

```
000252          PARS (ISP, 6) = M
000254          PARS (ISP, 7) = N
000256          PARS (ISP, 8) = MZ
000260          DO 4 K=1,20
000262          L=K+8
000264        4 PARS(ISP,L) = X(K)
000271          PARS (ISP,29) = NT1
000273          PARS (ISP,30) = NT2
000275          PARS (ISP,31) = NT3
000277          PARS (ISP,32) = NT4
000301          PARS (ISP,33) = NT5
000303          PARS (ISP,34) = NT6
000305          PARS (ISP,35) = NT7
000307          PARS (ISP,36) = NT8
000311          PARS (ISP,37) = NT9
000313          PARS (ISP,38) = NT10
000315          PARS ( ISP,39) = NT11
000317          PARS (ISP,40) = XEP1
000320          PARS (ISP,41) = XEP2
000322          PARS (ISP,42) = XEP3
000323          PARS (ISP,43) = NEXOP1
000325          PARS (ISP,44) = NEXOP2
000327          PARS (ISP,45) = NEXOP3
000331          PARS (ISP,46) = NEXOP4
000333          PARS (ISP,47) = NEXOP5
000335          IETS(ISP)=1
000337          GO TO 25
      C
000337       20 EPSI = PARS(ISP,1)
000341          RHOIN= PARS(ISP,2)
000343          THETA0 = PARS(ISP,3)
000344          RATIO  = PARS(ISP,4)
000346          TMMAX  = PARS(ISP,5)
000347          M = PARS(ISP,6)
000351          N = PARS(ISP,7)
000353          MZ= PARS(ISP,8)
000355          DO 21 K=1,20
000357          L= K+8
000361       21  X(K) = PARS(ISP,L)
000366          NT1  = PARS(ISP,29)
000370          NT2  = PARS(ISP,30)
000372          NT3  = PARS(ISP,31)
000374          NT4  = PARS(ISP,32)
000376          NT5  = PARS(ISP,33)
000400          NT6  = PARS(ISP,34)
000402          NT7  = PARS(ISP,35)
000404          NT8  = PARS(ISP,36)
000406          NT9  = PARS(ISP,37)
000410          NT10 = PARS(ISP,38)
000412          NT11 = PARS(ISP,39)
000414          XEP1 = PARS(ISP,40)
000415          XEP2 = PARS(ISP,41)
000417          XEP3 = PARS(ISP,42)
000420          NEXOP1=PARS(ISP,43)
000422          NEXOP2=PARS(ISP,44)
000424          NEXOP3=PARS(ISP,45)
000426          NEXOP4=PARS(ISP,46)
000430          NEXOP5=PARS(ISP,47)
```

```
000432          GO TO 25
          C
000433       25 CALL SETS(TMMAX)
000435          CALL TIMECS
000436          NPHASE=4                                                        000470
          C --- JUST TO GET AN INITIAL PRINTOUT                                 000480
000437          CALL EVALUS
000440          P0=0.0                                                          000500
000441          G=0.0                                                           000510
000442          H=0.0                                                           000520
000443          RSIGMA=0.0                                                      000530
000444          CALL OUTPUX (2)
000445          CALL STORES
000446          IF (NEXOP1.GT.1) CALL CHCKEX
000452          IF (NEXOP1.EQ.3) STOP 01072                                     000570
000456          IF (NEXOP1.EQ.5) STOP 01104                                     000580
000462          CALL FEASS
          C NPHASE=5 IS USED TO INDICATE NO FEASIBLE POINT EXIST                000600
000463          GO TO (30,30,30,30,10), NPHASE                                  000610
000474       30 NPHASE=2                                                        000620
000475          NICTR=0                                                         000630
000476          CALL BODYS
000477          DO 35 J=I,N
000501       35 V(J)=X(J)
000505          RETURN
          C                                                                     000670
          C     PARAMETER CARD                                                  000680
000505    50    FORMAT (5E12.0,3I4)                                             000690
          C     INITIAL  X VERCTOR CARD  FORMAT                                 000700
000505    60    FORMAT (6E12.6)                                                 000710
000505    70    FORMAT (6E20.7)                                                 000720
          C     OPTION CARD FORMAT                                              000730
000505    80    FORMAT (10I7)                                                   000750
000505    90    FORMAT(13H0 TOLERANCES  )                                       000760
000505    100   FORMAT (26H0 SECOND SET OF OPTIONS    )                         000770
000505    110   FORMAT (56H1 NONLINEAR PROGRAMMING ROUTINE-SUMT VERSION 4    SUB
               I          )                                                    000790
000505    120   FORMAT (1H0,5X,2HN=I3,6X,2HM=I3,6X,3HMZ=I3//8X,10HMAX. TIME=E14.7,  000800
               I4X,2HR=E14.7,4X,6HRATIO=E14.7,6X,8HEPSILON=E14.7,4X,6HTHETA=E14.7)  000810
000505    130   FORMAT (18H0 OPTIONS SELECTED)                                  000820
000505          END                                                             000830
```

```
                SUBROUTINE BODYS
          C
          C              OCTOBER 1970
          C
          C BODY COORDINATES THE FLOW AMONG THE SUBROUTINES THAT ACTUALLY DO THE
          C CALCULATIONS REQUIRED BY THE VARIOUS PARTS OF THE ALGORITHM.
000002          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000002          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000002          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000002          COMMON/CRSTS/DELX(100), DELX0(100), RHOIN,RATIO, EPSI, THETA0,
               1 RSIG1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), DOTT, PGRAD(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMINI, NPHASE, NSATIS
000002          COMMON/CONPAX / NF1, NF2,NF3
000002          NF2=2
000003          NF3=2
000004          MN=0
000005          NUMINI=0
          C     OPTION OF GETTING INITIAL RHO
000006          CALL RHOCOX
000007          CALL EVALUS
000010    10    CALL XMOVES
000011          GO TO (30,20), NT3
000017    20    CALL TIMECS
000020          CALL OUTPUX (1)
000022          GO TO 40
000023       30 CALL TCHECX
          C IN FEASIBILITY PHASE 4 MEANS  FEAS ACHIEVED
000024    40    GO TO (50,50,50,200), NSATIS
000034       50 CALL CONVRX (N1)
000036          GO TO (60,10,130), N1
          C  MINIMUM ACHIEVED IF  N1=1
000045    60    GO TO (70,80), NT3
000053    70    CALL TIMECS
000054          CALL OUTPUX (1)
          C --- NUMBER OF MINIMA ACHIEVED INCREASED BY 1
000056    80    NUMINI=NUMINI+1
000060          MN=0
000061          GO TO (190,90,90), NPHASE
000070    90    CALL ESTIMS
          C  FINAL RIGHT HAVE BEEN CALLED BY ESTIM-CONVERGED IF N2=1
000071          GO TO (100,110,120), NT4
          C   NT4=1 FINAL CONVERGENCE ON 0 ORDER ESTIMATES, NT4=2 CONVERGE ON FIRS
          C  ORDER ESTIMATES, NT4=3 CONVERGE ON SECOND ORDER ESTIMATES.
000100    100   CALL FINALS(NF1)
000102          GO TO (130,140), NF1
000110    110   GO TO (130,140), NF2
000116    120   GO TO (130,140), NF3
000124    130   RETURN
000125    140   RHO=RHO/RATIO
          C USING PREVIOSLY COMPUTED VALUES FOR F, AND RJ P IS RECOMPUTED WITH THE
          C NEW VALUE OF RHO.
000127          CALL PEVALX
          C A VECTOR IS LEFT IN DELX(I) BY ESTIM
000130          IF (NUMINI-2) 10,150,150
000133    150   GO TO (10,160,160), NT7
000142    160   CALL GRADS(2)
```

```
000144          CALL OPTS
000145          GO TO (180,170), NT3                                          001420
000153    170   WRITE (6,210)                    170 CONTINUE
000157          CALL OUTPUX (1)
000161    180   GO TO 50                                                     001450
000162    190   IF (G) 90,90,200                                             001460
000164    200   RETURN                                                       001470
          C --- DUAL VALUE GREATER THAN 0 MEANS NO FEASIBLE POINT EXISTS     001480
          C                                                                 001490
000165    210   FORMAT (6X,30HMOVED ON EXTRAPOLATION VECTOR )                001500
000165          END                                                          001510
```

```
      SUBROUTINE CHCKEX
C                                                                       001530
C             MARCH 1971                                                001540
C                                                                       001550
C CHCKER COMPUTES AND LIST THE FIRST PARTIAL DERIVATIVES USING GRAD1    001560
C AND THEN USING NUMERICAL DIFFERENCING (DIFF1).  IF REQUESTED THE      001570
C SECOND PARTIAL DERIVATIVES ARE COMPUTED AND LISTED USING MATRIX AND   001580
C DIFF2.                                                                001590
      COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
      COMMON /EQALS/H, H1, MZ
      MMZ=1+M+MZ                                                        001630
      DO 5 J=1,N                                                        001640
      DEL(J)=1.2345678                                                  001650
    5 CONTINUE                                                          001660
      DO 10 I=1,MMZ                                                     001670
      IN=I-1                                                            001680
      WRITE (6,170) IN                                                  001690
      CALL GRAD1S(IN)
      WRITE (6,180) (DEL(J),J=1,N)                                      001710
      CALL DIFF1S(IN)
      WRITE (6,180) (DEL(J),J=1,N)                                      001730
   10 CONTINUE                                                          001740
C...SOMETIMES ONLY FIRST DERIVATIVES ARE TO BE CHECKED                  001750
      IF (NEXOP1.LT.4) GO TO 160                                        001760
      WRITE (6,190)                                                     001770
      DO 150 I=1,MMZ                                                    001780
      IN=I-1                                                            001790
      WRITE (6,170) IN                                                  001800
      IT=2                                                              001810
      DO 30 K=1,N                                                       001820
      DO 20 J=1,N                                                       001830
   20 A(K,J)=0.                                                         001840
   30 CONTINUE                                                          001850
      CALL MATRXX(IN,IT)
      IF (IT.EQ.1) GO TO 150                                            001870
      DO 50 K=2,N                                                       001880
      KM1=K-1                                                           001890
      DO 40 J=1,KM1                                                     001900
      IF (A(K,J).EQ.0.0) GO TO 40                                       001910
      NEXOP1=5
      WRITE (6,210) K,J                                                 001930
      GO TO 60                                                          001940
   40 CONTINUE                                                          001950
   50 CONTINUE                                                          001960
   60 DO 90 K=1,N                                                       001970
      DO 70 J=K,N                                                       001980
      IF (A(K,J).NE.0.0) GO TO 80                                       001990
   70 CONTINUE                                                          002000
      WRITE (6,220) K                                                   002010
      GO TO 90                                                          002020
   80 WRITE (6,200) K,(A(K,J),J=1,N)                                    002030
   90 CONTINUE                                                          002040
      DO 110 K=1,N                                                      002050
      DO 100 J=1,N                                                      002060
  100 A(K,J)=0.                                                         002070
  110 CONTINUE                                                          002080
      WRITE(6,115) IN                                                   002090
```

```
000226   115   FORMAT (13H0 CALL DIFF2(,I2,1H) )                                  002100
000226         CALL DIFF2S(IN)
000230         DO 140 K=1,N                                                       002120
000232         DO 120 J=K,N                                                       002130
000233         IF (A(K,J).NE.0) GO TO 130                                         002140
000236   120   CONTINUE                                                           002150
000241         WRITE (6,220) K                                                    002160
000246         GO TO 140                                                          002170
000247   130   WRITE (6,200) K,(A(K,J),J=1,N)                                     002180
000266   140   CONTINUE                                                           002190
000271   150   CONTINUE                                                           002200
000274   160   CONTINUE                                                           002210
000274         RETURN                                                             002220
         C                                                                        002230
000275   170   FORMAT (   28H0CHECKER.....CONSTRAINT NO. , I3)                    002240
000275   180   FORMAT (1H0,24HCHECKER.....1ST PARTIALS/(1X,E20.8,E20.8,E20.8,E20.  002250
               18,E20.8,E20.8))                                                   002260
000275   190   FORMAT (1H0,24HCHECKER.....2ND PARTIALS)                           002270
000275   200   FORMAT (4H0ROW, I3      /(1X,E20.8,E20.8,E20.8,E20.8,E20.8,E20.8))  002280
000275   210   FORMAT (3H A(,I2,1H,,I2,10H) .NE. 0.0)                             002290
000275   220   FORMAT (4H ROW,I3,11H ALL ZEROS.)                                  002300
000275         END                                                                002310
```

```
                SUBROUTINE CONVRX (N1)
          C
          C              OCTOBER 1970
          C
          C AFTER EACH ITERATION OF THE ALGORITHM TO LOCATE THE MINIMUM OF THE
          C PENALTY FUNCTION, CONVRG DETERMINES IF THE CURRENT POINT IS CLOSE
          C ENOUGH TO THE POINT GIVING THE MINIMUM VALUE OF THE P FUNCTION.
          C     N1 SET EQUAL TO 1 IF MINIMUM HAS BEEN FOUND.
          C     N1 SET EQUAL TO 2 IF MINIMUM HAS NOT BEEN FOUND AND TIME IS NOT UP
          C     N1 SET EQUAL TO 3 OTHERWISE
          C     DOTT SET EQUAL TO (DEL P)(INVERSE(DEL(DEL P)))(DEL P) IN OPT
000003          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000003          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000003          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000003          COMMON/CRSTS/DELX(100), DELX0(100), RHOIN,RATIO, EPSI, THETA0,
               1 RSIG1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), DOTT, PGRAD(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMINI, NPHASE, NSATIS
000003          COMMON/EXPORX / NEXOP1, NEXOP2, XEP1, XEP2
000003          COMMON /TSWS/NSWW
000003          N1=2
000004          IF (MN.LE.1) Q1=P0
000011          GO TO (10,20,30), NT9
000020    10    IF (ABS(DOTT).LT.EPSI) GO TO 70
000024          GO TO 40
000024    20    IF (ABS(DOTT).LT.(P1-P0)/5.0) GO TO 70
000032          GO TO 40
000032    30    IF (ADELX.LT.EPSI) GO TO 70
000035    40    GO TO (50,60), NSWW
000043    50    IF (MN.LE.1) RETURN
000047          IF (P0+XEP2 .LT. Q1) GO TO 75
000053          WRITE (6,80)
000056          GO TO 70
000060    60    CALL PUNCHS
000061          WRITE (6,90)
000065          N1=3
          C
          C     FOUND THE MINIMUM TO THE SUBPROBLEM.
000067          RETURN
000070    70    N1=1
000071    75    Q1 = P0
000073          RETURN
          C
000073    80    FORMAT (100H APPARENTLY ROUNDOFF ERRORS PREVENT A MORE ACCURATE DE
               1TERMINATION OF THE MINIMUM OF THIS SUBPROBLEM.)
000073    90    FORMAT (48H0**** TIME IS UP, CALLING EXIT FROM CONVRG. ****)
000073          END
```

```
                SUBROUTINE DIFF1S(IN)
          C
          C             FEBUARY 1971
          C
          C DIFF1 COMPUTES THE FIRST DERIVATIVES BY NUMERICAL DIFFERENCING.
          C
          C--USER CAN CALL FOR DIFFERENCING OF SELECTED FUNCTIONS
000003          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000003          COMMON/EXPOPX / NEXOP1, NEXOP2, XEP1, XEP2
000003          COMMON/ STIRXS/XSTR( 100 )  ,  XSSS( 100), DDLL(100)
000003          DO 10 J=1,N
000005    10    XSTR(J)=X(J)
000011          DO 30 J=1,N
000012          IF (J.EQ.1) GO TO 20
000014          JM1=J-1
000015          X(JM1)=XSTR(JM1)
000017    20    X(J)=XSTR(J)+XEP1
000022          CALL RESTNX (IN,ZZ2)
000024          X(J)=XSTR(J)-XEP1
000027          CALL RESTNX (IN,ZZ1)
000032    30    DEL(J)=(ZZ2-ZZ1)/(2.*XEP1)
000042          X(N)=XSTR(N)
000043          RETURN
000044          END
```

```
                SUBROUTINE DIFF2S(IN)
          C
          C              OCTOBER 1970
          C
          C DIFF2 COMPUTES THE SECOND DERIVATIVES BY NUMERICAL DIFFERENCING.
          C
000003          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000003          COMMON/EXPOPX / NEXOP1, NEXOP2, XEP1, XEP2
000003          COMMON/ STIRXS/XSTR( 100 )  ,  XSSS( 100), DDLL(100)
000003          DO 10 J=1,N
000005    10    XSSS(J)=X(J)
000011          DO 50 J=1,N
000012          IF (J.EQ.1) GO TO 20
000014          JM1=J-1
000015          X(JM1)=XSSS(JM1)
000017    20    X(J)=XSSS(J)+XEP1
000022          CALL GRAD1S(IN)
000023          DO 30 I=1,N
000026    30    DDLL(I)=DEL(I)
000032          X(J)=XSSS(J)-XEP1
000035          CALL GRAD1S(IN)
000036          DO 40 I=J,N
000041    40    A(J,I)=(DDLL(I)-DEL(I))/(2.*XEP1)
000057    50    CONTINUE
000061          X(N)=XSSS(N)
000062          RETURN
000063          END
```

```
                SUBROUTINE ESTIMS
          C
          C              OCTOBER 1970
          C
          C ESTIM PERFORMS THE COMPUTATIONS TO ESTIMATE THE LAGRANGE MULTIPLIERS
          C AND MAKE THE FIRST- AND SECOND-ORDER ESTIMATES OF THE FINAL SOLUTION
          C OF THE PROBLEM.
000002          COMMON/SHARES/X(100), OEL(100), A(100,100),N,M, MN,NP1,NM1
000002          COMMON /EQALS/H, H1, MZ
000002          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000002          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000002          COMMON/CRSTS/OELX(100), DELX0(100), RHOIN,RATIO, EPSI, THETA0,
               1 RSIG1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), DOTT, PGRAO(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMINI, NPHASE, NSATIS
000002          COMMON/CONPAX / NF1, NF2,NF3
000002          CALL STORES
000003          Z10=RATIO**2
000005          Z9=RATIO
000006          Z1=1.0/Z9+1.0/Z10
000011          Z2=Z1+1./Z9**3
000014          Z3=1./Z9**3
000016          Z4=Z10+Z9
000017          Z5=Z9**3
000020          Z6=1.0/((Z10-1.0)*(Z9-1.0))
000024          Z7=1./Z9
000026          Z8=1./(Z9-1.)
000030          RQ=1.0/RHO
000032          IF (NUMINI-2) 150,80,10
000034    10    WRITE (6,330)                          10 CONTINUE
000040          P0=(PR2-Z4*PR1+Z5*P1)*Z6
000047          G=(RATIO*G1-GR1)/(RATIO-1.)
000053          DO 20 I=1,N
000055    20    X(I)=(XR2(I)-Z4*XR1(I)+Z5*X1(I))*Z6
000066          NP=NPHASE
000070          NPHASE=4
000071          CALL EVALUS
000072          NPHASE=NP
000074          CALL OUTPUX (2)
          C CHECK TO SEE IF ESTIMATES HAVE CONVERGED
000075          GO TO (70,30,70), NPHASE
000104    30    DO 50 J=1,M
000106          IF (RJ(J)) 40,50,50
000110    40    IF (THETA0+RJ(J)) 70,50,50
000113    50    CONTINUE
000116          GO TO (70,70,60), NT4
000125    60    CALL FINALS(NF3)
000127    70    CONTINUE
000127    80    WRITE (6,340)                          80 CONTINUE
000133          G=(RATIO*G1-GR1)/(RATIO-1.)
000140          P0=(Z9*P1-PR1)*Z8
000144          DO 90 I=1,N
000145    90    X(I)=(Z9*X1(I)-XR1(I))*Z8
000153          NP=NPHASE
000155          NPHASE=4
000156          CALL EVALUS
000157          NPHASE=NP
```

```
000161          CALL OUTPUX (2)
          C CHECK TO SEE IF ESTIMATES HAVE CONVERGED
000162          GO TO (140,100,140), NPHASE
000171    100   DO 120 J=1,M
000173          IF (RJ(J)) 110,120,120
000175    110   IF (RJ(J)+THETA0) 140,120,120
000200    120   CONTINUE
000203          GO TO (140,130,140), NT4
000212    130   CALL FINALS(NF2)
000214    140   CONTINUE
000214    150   WRITE (6,350)
000220          IF (M) 180,180,160
000222    160   DO 170 J=1,M
000224    170   RJ(J)=RHO/RJ1(J)
000230    180   IF (MZ) 210,210,190
000232    190   DO 200 J=1,MZ
000234          MNJ=M+J
000236    200   RJ(MNJ)=2.*RJ1(MNJ)*RO
000244    210   GO TO (220,240), NT2
000252    220   DO 230 I=1,N
000254    230   X(I)=RHO/X1(I)
000260      240 CALL OUTPUX (2)
000262          CALL REJECX
000263          IF (NUMINI-2) 280,300,250
000266    250   GO TO (280,310,260), NT7
          C     SECOND ORDER MOVE FOR NEXT MINIMUM
000275    260   DO 270 I=1,N
000277    270   DELX(I)=Z1*X1(I)-Z2*XR1(I)+Z3*XR2(I)
000310    280   PR2=PR1
000312          GR2=GR1
000313          PR1=P1
000314          GR1=G1
000315          DO 290 I=1,N
000317          XR2(I)=XR1(I)
000321    290   XR1(I)=X1(I)
000324          RETURN
000325    300   GO TO (280,310,310), NT7
000334    310   DO 320 I=1,N
000336    320   DELX(I)=(X1(I)-XR1(I))*Z7
000343          GO TO 280
          C
000344    330   FORMAT (/26H0     2ND  ORDER ESTIMATES )
000344    340   FORMAT (/26H0     1ST  ORDER ESTIMATES )
000344    350   FORMAT (/25H0  LAGRANGE MULTIPLIERS  )
000344          END
```

150 CONTINUE

```
                SUBROUTINE EVALUS
          C
          C              OCTOBER 1970
          C
          C IN THE NORMAL PHASE EVALU CALLS THE USER-SUPPLIEO ROUTINES TO EVALUATE
          C THE OBJECTIVE FUNCTION ANO THE CONSTRAINT FUNCTIONS AT THE CURRENT
          C POINT. IN THE FEASIBILITY PHASE THIS ROUTINE PUTS THE NEGATIVE SUM OF
          C THE VIOCATED CONSTRAINTS IN LOCATION F.
000002          COMMON/SHARES/X(100), OEL(100), A(100,100),N,M, MN,NP1,NM1
000002          COMMON /EQALS/H, H1, MZ
000002          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000002          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000002          COMMON/CRSTS/DELX(100), OELX0(100), RHOIN,RATIO, EPSI, THETA0,
               1 RSIG1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), OOTT, PGRAO(100), OIAG(100),
               3 PREV3'AOELX' NTCTR' NUMINI' NPHASE' NSATIS
000002          H=0.0
000003          RSIGMA=0.0
000004          F=0.0
000005          NSATIS=2
000006          GO TO (10,100,190,200), NPHASE
          C     =1  FEASIBILITY
          C     =2  NORMAL
          C     =3  GUESS
          C     =4  ALL FUNCTIONS ARE TO BE EVALUATEO
          C FEASIBILITY
000016    10    GO TO (20,40), NT2
          C NON-NEGATIVIES INCLUOED
000024    20    OO 30 I=1,N
000026          IF (X(I)) 260'260'30
000030    30    RSIGMA=RSIGMA-RHO*ALOG(X(I))
000040    40    IF (M.EQ.0) GO TO 90
000041          OO 80 J=1,M
000043          CALL RESTNX (J,RJ(J))
000045          IF (RJ1(J).LE.0.0) GO TO 50
000047          IF (RJ(J).GT.0.0) GO TO 60
          C VIOLATION OF A PREVIOUSLY SATISFIED CONSTRAINT
000052          GO TO 260
000052    50    IF (RJ(J).GT.0.0) GO TO 70
          C ALL VIOLATEO CONSTRAINTS ADDEO INTO OBJECTIVE FUNCTION
000055          F=F-RJ(J)
000057          GO TO 80
000057    60    RSIGMA=RSIGMA-RHO*ALOG(RJ(J))
000064          GO TO 80
          C INOICATES SATISFACTION OF CONSTRAINT(1ORMORE)
000065    70    NSATIS=1
000066          RSIGMA=RSIGMA-RHO*ALOG(RJ(J))
000073    80    CONTINUE
000076    90    CONTINUE
          C EQUALITIES NOT COMPUTEO IN FEAS. PHASE
000076          P0=F+RSIGMA
000100          G=F-RHO*FLOAT(M)
000105          IF (NT2.EQ.1) G=G-RHO*FLOAT(N)
000112          RETURN
          C  REGULAR  PHASE
000113    100   GO TO (110'130)' NT2
          C NON NEGATIVITIES INCLUOEO
```

```
000121   110    DO 120 I=1,N                                          004890
000123          IF (X(I)) 260,260,120                                 004900
000125   120    RSIGMA=RSIGMA-RHO*ALOG(X(I))                          004910
000135   130    IF (M.EQ.0) GO TO 150                                 004920
000136          DO 140 J=I+M                                          004930
000140          CALL RESTNX (J,RJ(J))
000142          IF (RJ(J).LE.0.0) GO TO 260                           004950
000144          RSIGMA=RSIGMA-RHO*ALOG(RJ(J))                         004960
000151   140    CONTINUE                                              004970
         C EVALUATE AND ADD IN EQUALITY CONSTRAINTS                   004980
000154   150    CONTINUE                                              004990
000154          CALL RESTNX ( 0,F)
000156          IF (MZ) 180,180,160                                   005010
000160   160    DO 170 I=1,MZ                                         005020
000162          J=I+M                                                 005030
000164          CALL RESTNX (J,RJ(J))
         C ADD INTO THIRD TERM OF P FUNCTION                          005050
000166          H=H+(RJ(J))**2                                        005060
000171   170    CONTINUE                                              005070
000174          H=H/RHO                                               005080
000175   180    P0=RSIGMA+H                                           005090
000177          P0=F+P0                                               005100
000201          G=2.*H-RHO*FLOAT(M)                                   005110
000205          G=G+F                                                 005120
000207          IF (NT2.EQ.1) G=G-RHO*FLOAT(N)                        005130
         C DUAL VALUE                                                 005140
000214          RETURN                                                005150
         C GUESS PHASE NOT CODED                                      005160
000215   190    RETURN                                                005170
         C---   STRAIGHT FUNCTION EVALUATION  ( MAIN+FEAS ONLY)       005180
000216   200    CONTINUE                                              005190
000216          IF (M.EQ.0) GO TO 220                                 005200
000217          DO 210 I=1,M                                          005210
000221          CALL RESTNX (I,RJ(I))
000223   210    CONTINUE                                              005230
000226     220  CALL RESTNX (0,F)
         C EQUALITY CONSTRAINTS                                       005250
000230          IF (MZ) 250,250,230                                   005260
000232   230    DO 240 I=1,MZ                                         005270
000234          KZ=M+I                                                005280
000236     240  CALL RESTNX ( KZ,RJ(KZ))
000243   250    RETURN                                                005300
         C CONSTRAINTS VIOLATED NOT SO BEFORE                         005310
000244   260    NSATIS=3                                              005320
000245          P0=10.E35                                             005330
000247          RETURN                                                005340
000247          END                                                   005350
```

```
                SUBROUTINE FEASS
          C
          C              OCTOBER 1970
          C
          C FEAS DETERMINES WHETHER THE STARTING POINT IS FEASABLE. IF IT IS NOT,
          C FEAS LOOKS FOR A FEASABLE ONE. IF NONE EXISTS, A MESSAGE IS PRINTED
          C AND CONTROL RETURNS TO MAIN.
000002          COMMON/SHARES/X(100), OEL(100), A(100,100),N,M, MN,NP1,NM1
000002          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000002          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000002          COMMON/CRSTS/DELX(100), OELX0(100), RHOIN,RATIO, EPSI, THETA0,
               1 RSIG1, G1,     X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), OOTT, PGRAO(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMINI, NPHASE, NSATIS
000002          NPHASE=1
000003          GO TO (10,50), NT2
000011    10    NFIX=1
000012          OO 30 I=1,N
000014          IF (X(I)) 20,20,30
000016    20    NFIX=2
000017          X(I)=1.E-05
000021    30    CONTINUE
000024          GO TO (50,40), NFIX
000032    40    NPHASE=4
000033          CALL EVALUS
          C JUST GET ALL CONSTRAINTS EVALUATEO
000034          NPHASE=1
000035          WRITE (6,130)
000041          CALL OUTPUX( 2)
000043    50    IF (M) 90,90,60
000045    60    DO 70 I=1,M
000047          IF (RJ(I)) 100,100,70
000051    70    CONTINUE
000054    80    CALL TIMECS
000055          WRITE (6,140)
000061          G=0.0
000062          CALL RESTNX ( 0,F)
000064          CALL OUTPUX (2)
000066    90    RETURN
000067    100   CALL BODYS
000070          OO 110 I=1,M
000072          IF (RJ(I)) 120,120,110
000074    110   CONTINUE
000077          GO TO 80
000077    120   WRITE (6,150)                          120 CONTINUE
          C TO INDICATE TO MAIN TO START ON NEXT PROBLEM.
000103          NPHASE=5
000104          GO TO 90
          C
000105    130   FORMAT (1H0,2X,48HMADE VIOLATED NON-NEGATIVITIES SLIGHTLY POSITIVE
               1)
000105    140   FORMAT (51H0*****THE FEASIBLE STARTING POINT TO BE USED IS ...)
000105    150   FORMAT (3X,89HTHIS PROBLEM POSSESSES NO FEASIBLE STARTING POINT, W
               1ILL LOOK FOR DATA FOR NEXT PROBLEM.  )
000105          ENO
```

```
                SUBROUTINE FINALS(N2)
          C
          C                OCTOBER 1970
          C
          C FINAL CONTAINS THE TESTS USED TO DETERMINE WHETHER A POINT SATISFIES
          C THE FINAL CONVERGENCE CRITERION CHOOSEN TO DETERMINE IF THE NLP
          C PROBLEM HAS BEEN SOLVED.
          C N2 SET EQUAL TO 1 IF CONVERGENCE CRITERION IS SATISFIED.
          C N2 SET EQUAL TO 2 OTHERWISE.
000003          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000003          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000003          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000003          COMMON/CRSTS/DELX(100), DELX0(100), RHOIN,RATIO, EPSI, THETA0,
               1 RSIG1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), DOTT, PGRAD(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMINI, NPHASE, NSATIS
000003          GO TO (10,20,30), NT5
000012    10    EPSIL=ABS(F/G-1.)
000016          IF (EPSIL-THETA0) 50,50,70
000021    20    IF (ABS(RSIGMA)-THETA0) 50,50,70
000025    30    IF (NUMINI-1) 50,40,40
000030    40    PEST=PR1-(PR1-P0)/(1.-1./SQRT(RATIO))
000040          EPSIL=ABS(PEST/G-1.)
000043          IF (EPSIL-THETA0) 50,70,70
000046    50    N2=1
000047          GO TO (80,60), NT6
000055    60    CALL PUNCHS
000056          GO TO 80
000060    70    N2=2
000061    80    RETURN
000062          END
```

```
                SUBROUTINE GRADS(IS)
          C
          C                OCTOBER 1970
          C
          C GRAD COMPUTES THE GRADIENT OF THE PENALTY FUNCTION AND THE OUTER
          C PRODUCT FACTORS OF THE MATRIX OF SECOND PARTIALS OF P.
          C  IF  (IS=1) ACCUM. MATRIX OF 2ND PARTIALS  IF(IS=2) OUNT
000003          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000003          COMMON /EQALS/H, H1, MZ
000003          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000003          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000003          COMMON/CRSTS/DELX(100), DELX0(100), RHOIN,RATIO, EPSI, THETA0,
               1 RSIG1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), DOTT, PGRAD(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMINI, NPHASE, NSATIS
000003          GO TO (10,30), IS
000011    10    DO 20 I=1,N
000013          DO 20 J=1,I
000014    20    A(I,J)=0.
000024    30    DO 40 I=1,N
000026    40    DELX0(I)=0.
          C THIS SECTION WORKS CORRECTLY IN FEASIBILITY PHASE AS WELL AS NORMAL PH
000031          GO TO (50,80), NT2
000037    50    DO 70 I=1,N
000041          DELX0(I)=-RHO/X(I)
000044          GO TO (60,70), IS
000051    60    A(I,I)=(-DELX0(I)/X(I))
000056    70    CONTINUE
000061    80    CONTINUE
000061          IF (M.LE.0) GO TO 180
000063          DO 170 K=1,M
000064          CALL GRADIS(K)
000065          IF (RJ(K).GT.0.0) GO TO 110
          C ALL VIOLATED CONSTRAINT  GRADS ADDED TO OBJ. FUNCTION
000071          DO 100 I=1,N
000072          IF (DEL(I)) 90,100,90
000073    90    DELX0(I)=DELX0(I)-DEL(I)
000076    100   CONTINUE
000101          GO TO 170
000101    110   TT=RHO/RJ(K)
000104          DO 160 I=1,N
000105          IF (DEL(I)) 120,160,120
          C IF DEL(I)=0 SKIP ALL THE FOLLOWING COMPUTATION INVOLVING * BY DEL(I)
000106    120   T=TT*DEL(I)
000111          DELX0(I)=DELX0(I)-T
000113          GO TO (130,160), IS
000120    130   T=T/RJ(K)
000122          DO 150 JJ=1,I
000124          IF (DEL(JJ)) 140,150,140
000125    140   A(I,JJ)=A(I,JJ)+T*DEL(JJ)
000133    150   CONTINUE
000136    160   CONTINUE
000141    170   CONTINUE
          C  EQUALITY CHANGES FOR GRAD
000144    180   IF (MZ.LE.0) GO TO 250
000146          GO TO (250,190,250), NPHASE
000155    190   RQ=2./RHO
```

```
000157          DO 240 J=1,MZ                                          006790
000161          K=M+J                                                  006800
000163          CALL GRAD1S(K)
000164          TT=RQ*RJ(K)                                            006820
000167          DO 230 I=1,N                                           006830
000171          IF (DEL(I).EQ.0.0) GO TO 230                           006840
000172          DELX0(I)=DELX0(I)+DEL(I)*TT                            006850
000175          GO TO (200,230), IS                                    006860
000203    200   T=RQ*DEL(I)                                            006870
000206          DO 220 JJ=I,I                                          006880
000207          IF (DEL(JJ)) 210,220,210                               006890
000210    210   A(I,JJ)=A(I,JJ)+T*DEL(JJ)                             006900
000216    220   CONTINUE                                               006910
000221    230   CONTINUE                                               006920
000224    240   CONTINUE                                               006930
000226    250   GO TO (260,280), IS                                   006940
000234    260   DO 270 I=I,N                                           006950
000236    270   DIAG(I)=A(I,I)                                         006960
000246    280   GO TO (290,330,290), NPHASE                            006970
        C LEAVES NEGATIVE GRADIENT IN DELP                             006980
000255    290   DO 300 I=I,N                                           006990
000257    300   DELX0(I)=-DELX0(I)                                     007000
000263    310   ADELX=0.                                               007010
000264          DO 320 I=1,N                                           007020
000266    320   ADELX=ADELX+DELX0(I)**2                                007030
000273          ADELX=SQRT(ADELX)                                      007040
000275          RETURN                                                 007050
000275    330   CALL GRAD1S(0)
000277          DO 340 I=1,N                                           007070
000302    340   DELX0(I)=-DELX0(I)-DEL(I)                              007080
        C LEAVES THE NEG. GRAD OF P IN DELX0                           007090
000307          GO TO 310                                              007100
000307          END                                                    007110
```

```
                SUBROUTINE INVERX(NSME)
          C
          C              OCTOBER 1970
          C
          C INVERS SOLVES THE SET OF EQUATIONS FOR THE MOVE-VECTOR USING THE
          C CROUT PROCEDURE. IF THE MATRIX IS NOT POSITIVE DEFINITE, A DIFFERENT
          C METHOD IS USED.
          C
          C****PERFORMING A L-U DECOMPOSITION OF THE MATRIX A, TAKING ADVANAGE OF
          C****THE SYMMETRY OF THE A MATRIX.
          C****IF A NON-POSITIVE PIVOT CANDIDATE IS GENERATED, THEN MCCORMICK,S
          C****PROCEDURE IS USED(SEE PP. 167-168 IN FIACCO AND MCCORMICK).
          C
          C****IF NSME =1 WORKING WITH A NEW A MATRIX.  IF NSME= 2 USING PREVIOUS
          C****A MATRIX, BUT HAVE A NEW RIGHT-HAND-SIDE.
          C
          C****NINV IS THE NUMBER OF NON-POSITIVE PIVOT CANDIDATES GENERATED.
000003          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000003          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000003          COMMON/CRSTS/DELX(100), DELX0(100), RHOIN,RATIO, EPSI, THETA0,
               1 RSIG1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), DOTT, PGRAD(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMINI, NPHASE, NSATIS
000003          COMMON/EXPORX / NEXOP1, NEXOP2, XEP1, XEP2
000003          DIMENSION B(100)
000003          EQUIVALENCE (B,DELX)
000003          GO TO(20, 170), NSME
000011    20    NINV=0
000012          IF (A(1,1)) 40,30,50
000014    30    NINV=1
000015          GO TO 70
000016    40    NINV=1
000017    50    A(1,1)=1./A(1,1)
000021          DO 60 I=2,N
000022    60    A(1,I)=A(1,I)*A(1,1)
000030    70    DO 160 J=2,N
000032          JM1=J-1
000034          T=0.
000035          DO 90 I=1,JM1
000036          IF (A(I,J)) 80,90,80
000041    80    T=T+A(J,I)*A(I,J)
000051    90    CONTINUE
000054          A(J,J)=A(J,J)-T
000060          IF (A(J,J)) 110,100,120
000064   100    NINV=NINV+1
000066          GO TO 170
000066   110    NINV=NINV+1
000070   120    A(J,J)=1./A(J,J)
000074          IF (J.EQ.N) GO TO 170
000076          JP1=J+1
000100          DO 150 L=JP1,N
000101          T=0.
000102          DO 140 I=1,JM1
000104          IF (A(I,J)) 130,140,130
000107   130    T=T+A(L,I)*A(I,J)
000117   140    CONTINUE
000122          A(L,J)=A(L,J)-T
```

```
000127          A(J,L)=A(L,J)*A(J,J)                                    007690
000136    150   CONTINUE                                                007700
000140    160   CONTINUE                                                007710
000142    170   CONTINUE                                                007720
000142          IF (NINV) 180,180,290                                   007730
000144    180   B(1)=B(1)*A(1,1)                                        007740
000146          DO 210 J=2,N                                            007750
000147          T=0.                                                    007760
000150          JM1=J-1                                                 007770
000152          DO 200 I=1,JM1                                          007780
000153          IF (A(J,I)) 190,200,190                                 007790
000156    190   T=T+A(J,I)*B(I)                                         007800
000164    200   CONTINUE                                                007810
000167          B(J)=(B(J)-T)*A(J,J)                                    007820
000174    210   CONTINUE                                                007830
000177          DO 240 I=1,NM1                                          007840
000200          NMK=N-I                                                 007850
000202          DO 230 J=1,I                                            007860
000203          L=NP1-J                                                 007870
000205          IF (A(NMK,L)) 220,230,220                               007880
000210    220   B(NMK)=B(NMK)-A(NMK,L)*B(L)                             007890
000216    230   CONTINUE                                                007900
000221    240   CONTINUE                                                007910
000223    250   GO TO (280,260), NT3                                    007920
000231    260   WRITE (6,430)                                           260 CONTINUE
000235          WRITE (6,420) (DELXO(I),I=1,N)
000250    270   WRITE (6,440)                                           270 CONTINUE
000254          WRITE (6,420) (DELX(I),I=1,N)
000267    280   RETURN                                                  007970
          C---  COMPUTE ORTHOGONAL MOVE                                 007980
000270    290   CONTINUE                                                007990
000270          DO 350 II=1,N                                           008000
000273          I=N-II+1                                                008010
000276          IF (A(I,I)) 310,300,320                                 008020
000302    300   B(I)=0.0                                                008030
000304          GO TO 350                                               008040
000304    310   B(I)=1.0                                                008050
000306          GO TO 330                                               008060
000307    320   B(I)=0.0                                                008070
000311    330   IP1=I+1                                                 008080
000313          IF (IP1.GT.N) GO TO 350                                 008090
000316          DO 340 J=IP1,N                                          008100
000317    340   B(I)=B(I)-A(I,J)*B(J)                                   008110
000331    350   CONTINUE                                                008120
000334          GO TO 360                                               008130
          C-- CHECK MAYBE DO DIFF FOR P.S.D.                            008140
000334    360   ZC2=0.0                                                 008150
000335          DO 370 I=1,N                                            008160
000337    370   ZC2=ZC2+DELXO(I)*B(I)                                   008170
000344          IF (ZC2) 380,400,400                                    008180
000345    380   DO 390 I=1,N                                            008190
000347    390   B(I)=-B(I)                                              008200
000353    400   WRITE (6,450)
          C MCC ZANGWILL  ONE MOD                                       008220
000357          IF (NEXOP2.NE.2) GO TO 250                              008230
000362          DO 410 K=1,N                                            008240
000364    410   B(K)=B(K)+DELXO(K)                                      008250
000371          GO TO 250                                               008260
```

```
          C
000371   420   FORMAT (7E17.8)                                                  008270
000371   430   FORMAT (1H0,6X,12HDEL P VECTOR)                                  008280
000371   440    FORMAT(1H0,6X,24HSECOND ORDER MOVE VECTOR)                      008290
000371   450   FORMAT (1H0,6X,15HORTHOGONAL MOVE)                               008300
000371         END                                                               008310
                                                                                 008320
```

```
                SUBROUTINE OPTS
          C
          C              MARCH 1971
          C
          C OPT LOOKS FOR A MINIMUM ALONG THE SEARCH VECTOR USING THE GOLDEN
          C SECTION SEARCH METHOD.
000002          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000002          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000002          COMMON/CRSTS/DELX(100), DELX0(100), RHO1N,RAT10, EPSI, THETA0,
               1 RSIG1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), DOTT, PGRA0(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMIN1, NPHASE, NSAT1S
000002          KSW=1
000003          N405=1
000004          P31=P0
000006          ISW=1
000007          DOTT=0.
000010          DO 10 J=1,N
000011    10    DOTT=DOTT+DELX(J)*DELX0(J)
000016          GO TO 40
000017    20    DO 30 I=1,N
000021    30    DELX(I)=-DELX(I)
000025    40    CONTINUE
000025          N404=0
000026          MN=MN+1
          C MN IS NOW NUMB. OF POINTS AFTER MIN ACHIEVED
000030          NTCTR=NTCTR+1
000031          DO 50 I=1,N
000032    50    X2(I)=X(I)
000036          PX1=P0
000037          N401=0
000040    60    N401=N401+1
000042          DO 70 I=1,N
000043    70    X(I)=X2(I)+DELX(I)
000050          CALL EVALUS
          C 1 MEANS SATIS.OF CONSTRAINT NT.PREV. 2MEANS NOCHANGE 3MEANS VIOLATION
          C IF POINT IS NOT FEASIBLE GIVE IT AN ARBRITRARILY HIGH VALUE
000051          GO TO (540,90,80), NSAT1S
000060    80    PX2=10.E35
000062          P0=10.E35
000063          GO TO 100
000064    90    CONTINUE
000064          PX2=P0
000066          IF (PX1-PX2) 100,100,150
000070   100    IF (N401-2) 130,110,110
000073   110    DO 120 I=1,N
000075   120    X1(I)=X(I)
000101          P1=PX2
000102          GO TO 430
          C ONLY ONE  POINT  SO FAR COMPUTED
000103   130    DO 140 I=1,N
000105   140    X3(I)=X2(I)
000111          PREV3=PX1
000112          GO TO 180
000113   150    DO 160 I=1,N
000115          X3(I)=X2(I)
000117          X2(I)=X(I)
```

```
000120   160   DELX(I)=1.61803399*DELX(I)                                  008900
000124         PREV3=PX1                                                   008910
000126         PX1=PX2                                                     008920
000127         GO TO 60                                                    008930
         C GOLDEN SECTION SEARCH METHOD.                                   008940
         C B VECTOR  GOES TO X1(I)                                         008950
000127   170   P0=1.E36                                                    008960
000131         N404=N404+1                                                 008970
000132   180   DO 190 I=1,N                                                008980
000134   190   X1(I)=X(I)                                                  008990
000140         P1=P0                                                       009000
000141         DO 200 I=1,N                                                009010
000143         X(I)=.38196601*(X1(I)-X3(I))+X3(I)                          009020
000150   200   X2(I)=X(I)                                                  009030
000153         CALL EVALUS
000154         GO TO (540,270,210), NSATIS                                 009050
000163   210   IF (N404.LT.30) GO TO 170                                   009060
000166   211   CONTINUE                                                    009070
         C THERE IS NO REFERENCE TO 211, THE ABOVE STATEMENT IS A DUMMY STATEMENT 009071
         C--IT IS POSSIBLE NO FEASIBLE POINT EXIST, IF NOT TRY MOVING ON DELXO. 009080
         C--  IF IT IS NOT POSSIBLE TO MOVE ON DELXO THEN WE MUST BE AT A  009090
         C--  SOLUTION OF NLP PROBLEM.                                     009100
000166         IF (N404.GT.100) GO TO 240                                  009110
000172   220   DO 230 I=1,N                                                009120
000174         IF (ABS(ABS(X3(I)/X1(I))-1.).GT.1.E-7) GO TO 170            009130
000203   230   CONTINUE                                                    009140
000205   240   GO TO (250,260), N405                                       009150
000213   250   N405=2                                                      009160
         C--TRY TO MOVE ON GRADIENT.                                       009170
000214         NTCTR=NTCTR-1                                               009180
000216         MN=MN-1                                                     009190
000217         GO TO 20                                                    009200
000217   260   WRITE (6,580)                  260 CONTINUE
000223         CALL TIMECS
000224         CALL OUTPUX(1)
000226         CALL  REJECA
000227         STOP 22042                                                  009250
         C                                                                 009260
000231   270   CONTINUE                                                    009270
000231         N404=0                                                      009280
000232         PX1=P0                                                      009290
000234         DO 280 I=1,N                                                009300
000235   280   X(I)=0.38196601*(X1(I)-X2(I))+X2(I)                         009310
000244         CALL EVALUS
000245         GO TO (540,290,220), NSATIS                                 009330
000254   290   PX2=P0                                                      009340
000256         N401=1                                                      009350
000257   300   N401=N401+1                                                 009360
000261         IF (N401-25) 340,310,310                                    009370
000263   310   KSW=2                                                       009380
000264         IF (N401-40) 320,460,460                                    009390
000267   320   DO 330 I=1,N                                                009400
000271         IF (ABS(X2(I)/X(I)-1.0).GE.1.E-7) GO TO 340                 009410
000276   330   CONTINUE                                                    009420
000301         GO TO 460                                                   009430
000301   340   IF (ABS(PX1/PX2-1.).LE.1.E-7) GO TO 460                     009440
000306         IF (PX1-PX2) 350,460,400                                    009450
         C FROM LEFTORIGHT  X3(I)(PREV3)X2(I)(PX1)X(I)PX2 X1(I)P1          009460
```

```
000310   350    DO 360 I=I,N                                          009470
000312   360    X1(I)=X(I)                                            009480
         C THROW AWAY RIGHT PART                                      009490
000316          P1=PX2                                                009500
000317          DO 370 I=1,N                                          009510
         C POINTXP1 BECOMES XP2                                       009520
00032I   370    X(I)=.38196601*(X1(I)-X3(I))+X3(I)                    009530
         C TEMPORARILY IN  X STORAGE                                  009540
000330          CALL EVALUS
00033I          GO TO (540,380,170), NSATIS                           009560
000340   380    CONTINUE                                              009570
000340          PX2=PXI                                               009580
         C SWITCH VECTORS TO PROPER POSITION                          009590
000342          PXI=P0                                                009600
000343          DO 390 I=I,N                                          009610
000344          XX=X2(I)                                              009620
000346          X2(I)=X(I)                                            009630
000347   390    X(I)=XX                                               009640
000353          GO TO 300                                             009650
         C LEFT SIDE  TOSSED  AWAY                                    009660
         C-- CHANGES FOR NONUNIMODAL FN                               009670
         C-- GO TO THROW AWAY RIGHT IN THIS CASEINIT VAL LT FIB PT    009680
000353   400    IF (PREV3-PX2) 350,350,410                            009690
000356   410    DO 420 I=1,N                                          009700
000360          X3(I)=X2(I)                                           009710
000362   420    X2(I)=X(I)                                            009720
000365          PREV3=PX1                                             009730
000367          PXI=PX2                                               009740
000370   430    DO 440 I=I,N                                          009750
000372   440    X(I)=0.38196601*(XI(I)-X2(I))+X2(I)                   009760
00040I          CALL EVALUS
000402          GO TO (540,450,170), NSATIS                           009780
0004II   450    CONTINUE                                              009790
0004II          PX2=P0                                                009800
000413          GO TO 300                                             009810
         C THE INTERIOR POINTS NOW GIVE EQUAL VALUE FOR P. COMPUTE MIDPOINT.  009820
000413   460    DO 470 I=I,N                                          009830
000415          OELX0(I)=X(I)                                         009840
000417          X(I)=(OELX0(I)+X2(I))*0.5                             009850
000422   470    CONTINUE                                              009860
000424          CALL EVALUS
000425          GO TO (480,490), KSW                                  009880
000433   480    IF (ABS(P0/PX1-1.).GT.1.E-7) GO TO 520                009890
00044I   490    GO TO (500,5I0), ISW                                  009900
000447   500    IF (P0.LT.P3I) GO TO 510                              009910
000452          ISW=2                                                 009920
         C IF P-FUNCTION DION.T  GO DOWN TRY NEG VECT.                009930
000453          GO TO 20                                              009940
000453   510    RETURN                                                009950
000454   520    DO 530 I=I,N                                          009960
000456   530    X(I)=DELX0(I)                                         009970
000462          GO TO 350                                             009980
         C ARE  WE NOW IN FEASIBILITY PHASE                           009990
000462   540    DO 550 I=I,M                                          010000
000464          IF (RJ(I)) 560,560,550                                0100I0
000466   550    CONTINUE                                              010020
00047I          NSATIS=4                                              010030
000472          RETURN                                                 0I0040
```

```
        C---  PROBLEM HAS BECOME FEASIBLE                                    010050
        C ---  P - FUNCTION CHANGES IF A CONSTRAINT BECOMES FEASIBLE         010060
000472  560   MN=0                                                           010070
000473        DO 570 I=1,M                                                   010080
000475  570   RJ1(I)=RJ(I)                                                   010090
000501        RETURN                                                         010100
        C                                                                    010110
000501  580   FORMAT ( 80H  OPT CAN*T FIND A FEASIBLE POINT,THAT GIVES A LOWER V  010120
             1ALUE OF THE P-FUNCTION.          )                             010130
000501        END                                                            010140
```

```
                SUBROUTINE OUTPUX (K)
        C                                                                       010160
        C               OCTOBER 1970                                            010170
        C                                                                       010180
        C OUTPUT PRINTS OUT INFORMATION ON THE RESULTS OF EACH ITERATION AND THE 010190
        C SOLUTION ESTIMATES AND THE ESTIMATES OF THE LAGRANGE MULTIPLIERS      010200
000003          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000003          COMMON /FGALS/H, H1, MZ
000003          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000003          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000003          COMMON/CRSTS/DELX(100), OELX0(100), RHO1N,RATIO, EPS1, THETA0,
               1 RSIG1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), DOTT, PGRAO(100), DIAG(100),
               3 PREV3'ADELX' NTCTR' NUMINI' NPHASE' NSATIS
000003          NZ=M+MZ                                                         010290
000005          GO TO (10,20), K                                                010300
000013    10    WRITE (6,60)                                                    010310
000017          WRITE (6,70) NTCTR,DOTT,RHO,ADELX,NPHASE                        010320
000035    20    WRITE (6'80) F'P0'G'RSIGMA'H                                    010330
000053          WRITE (6,90) (X(I),I=1,N)                                       010340
000066          WRITE (6,110)                                                   010350
000072          GO TO (30,40), NT2                                              010360
000101    30    WRITE (6,120)                                                   010370
000105          WRITE (6,100) (RJ(I),I=1,NZ)                                    010380
000120          GO TO 50                                                        010390
000122    40    WRITE (6,100) (RJ(I),I=1,NZ)                                    010400
000135    50    RETURN                                                          010410
        C                                                                       010420
000136    60    FORMAT (50H0*********************************************      )010430
000136    70    FORMAT (10X,6HPOINT=I4,6X,6H DOTT=E15.7,6X,4HRHO=E15.7,6X,10HMAGNI 010440
               1TUDE=E15.7,6X,6HPHASE=I2)                                       010450
000136    80    FORMAT (8X,2HF=E15.7,5X,2HP=E15.7,5X,2HG=E15.7,5X,7HRSIGMA=E15.7,5 010460
               1X'2HH=E15.7)                                                    010470
000136    90    FORMAT (6X,25HTHE CURRENT VALUE OF X IS/(6E20.7))               010480
000136   100    FORMAT (6E20.7)                                                 010490
000136   110    FORMAT (6X,21HTHE CONSTRAINT VALUES)                            010500
000136   120    FORMAT (28X'34HNOT INCLUDING THE NON-NEGATIVITIES)              010510
000136          END
```

```
                SUBROUTINE OUTPUX (K)
        9999    CONTINUE
                RETURN
                END
```

```
                SUBROUTINE PEVALX
          C
          C              OCTOBER 1970
          C
          C PEVALU COMPUTES THE VALUE OF THE PENALTY FUNCTION AND THE VALUE OF THE
          C DUAL USING PREVIOUSLY COMPUTED VALUES FOR F, AND RJ.
000002          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000002          COMMON /EQALS/H, H1, MZ
000002          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000002          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000002          COMMON/CRSTS/DELX(100), DELX0(100), RHOIN,RATIO, EPS1, THETA0,
               1 RSIG1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), DOTT, PGRAD(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMINI, NPHASE, NSATIS
000002          H=0.0
000003          RSIGMA=0.0
          C NONNEGS IF INCLUDED ARE ADDED TO  P--ARE POS, IN ALL PHASES
000004          GO TO (10,30), NT2
000012    10    DO 20 I=1,N
000014    20    RSIGMA=RSIGMA-RHO*ALOG(X(I))
000024    30    GO TO (40,50,150), NPHASE
          C OBJECTIVE FUNCTION   - SIGMA VIOL. CONSTS.
000033    40    F=0.0
000034    50    IF (M) 100,100,60
000036    60    DO 90 J=1,M
000040          IF (RJ(J)) 80,80,70
000042    70    RSIGMA=RSIGMA-RHO*ALOG(RJ(J))
000047          GO TO 90
000050    80    F=F-RJ(J)
000053    90    CONTINUE
          C EQUALITIES NOT ADDED IN FEAS. PHASE
000056    100   CONTINUE
000056          IF (MZ) 140,140,110
000060    110   GO TO (140,120,150), NPHASE
000067    120   DO 130 I=1,MZ
000071          K=M+I
000073    130   H=H+RJ(K)**2
000100          H=H/RHO
000101    140   HS=H+RSIGMA
000103          P0=F+HS
000105          HMS=2.*H-RHO*FLOAT(M)
000111          G=F+HMS
000113          IF (NT2.EQ.1) G=G-RHO*FLOAT(N)
000120    150   RETURN
000121          END
```

```
                SUBROUTINE PUNCHS
          C
          C              OCTOBER 1970
          C
          C THIS SUBROUTINE PUNCHES THE STOPPING POINTS AND ASSOCIATED PARAMETERS
          C SO THAT ANOTHER RUN MAY BE MADE STARTING WHERE THE CURRENT ONE
          C STOPPED
          C  THIS ROUTINE IS CALLED IF NT6=2.
000002          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000002          COMMON /EQALS/H, H1, MZ
000002          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000002          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000002          COMMON/CRSTS/DELX(100), DELX0(100), RHOIN,RATIO, EPSI, THETA0,
               1 RSIG1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), DOTT, PGRAD(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMINI, NPHASE, NSATIS
000002          COMMON/EXPOPX / NEXOP1, NEXOP2, XEP1, XEP2
000002          T=60.0
000004          WRITE (7,10) EPSI,RHO,THETA0,RATIO,T,M,N,MZ
          C  TMMAX.IS SET TO 60. SECONDS
000027          NT1=3
000030          WRITE (7,20) (X(I),I=1,N)
          C SET  RHO OPTION SO THIS VALUE OF RHO WILL BE USE FOR THE RESTART.
000043          WRITE (7,30) NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000073          WRITE (7,20) XEP1, XEP2
000103          WRITE (7,30) NEXOP1,NEXOP2
000113          RETURN
          C
000114    10    FORMAT (5E12.5,3I4)
000114    20    FORMAT (6E12.5)
000114    30    FORMAT (10I7)
000114          END
```

```
                SUBROUTINE REJECX
          C
          C              OCTOBER 1970
          C
          C REJECT RETURNS THE STORED VALUES OF THE OBJECTIVE FUNCTION, THE
          C CONSTRAINT FUNCTIONS AND THE PENALTY FUNCTION TO THEIR NORMAL LOCATION
000002          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000002          COMMON /EQALS/H, H1, MZ
000002          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000002          COMMON/CRSTS/DELX(100), DELX0(100), RHO1N,RATIO, EPS1, THETA0,
               1 RS1G1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), DOTT, PGRAD(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUM1N1, NPHASE, NSATIS
000002          DO 10 I=1,N
000004    10    X(I)=X1(I)
000010          MMZ=M+MZ
000012          DO 20 J=1,MMZ
000013    20    RJ(J)=RJ1(J)
000017          P0=P1
000020          RSIGMA=RSIG1
000022          G=G1
000023          F=F1
000025          H=H1
000026          RETURN
000027          END
```

```
                SUBROUTINE RHOCOX
          C
          C              OCTOBER 1970
          C
          C     SUBROUTINE  TO COMPUTE INITIAL RHO VALUE
          C     CONTROLLED  BY  COL. 7  ON OPTION CARD
000002          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000002          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000002          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000002          COMMON/CRSTS/DELX(100), DELX0(100), RHOIN,RATIO, EPS1, THETA0,
               1 RSIG1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), DOTT, PGRAD(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMINI, NPHASE, NSATIS
000002          GO TO (110,50,10,190), NT1
000012    10    RHO=RHOIN
000014    20    IF (RHO) 30,30,40
000016    30    RHO=1.
000020    40    RETURN
000021    50    NPAR1=1
000022    60    RHO=1.
          C     2 MEANS  RHO WHICH  MINIMIZES GRADIENT  MAG.
000024          CALL GRADS(2)
000025          DO 70 I=1,N
000027    70    PGRAD(I)=DELX0(I)
000033          RHO=2.
000034          CALL GRADS(2)
000036          DO 80 I=1,N
000040          DELX0(I)=DELX0(I)-PGRAD(I)
000042    80    PGRAD(I)=PGRAD(I)-DELX0(I)
000046          GO TO (90,130), NPAR1
000054    90    DOT1=0.
000055          DOT2=0.
000056          DO 100 I=1,N
000057          DOT1=DOT1+DELX0(I)*PGRAD(I)
000062    100   DOT2=DOT2+DELX0(I)**2
000067          RHO=ABS(DOT1/DOT2)
000071          GO TO 20
          C  3 MEANS COMPUTE RHO SO AS TO MINIMIZE DEL P(/DDP/1.)DEL P
000072    110   NPAR2=1
000073    120   NPAR1=2
          C     USE  OF  AND OR  SUBROUTINE
000074          GO TO 60
000075    130   RHO=1.
          C     ASSUME   SIGMA TERM IS CONSID. GRTER THAN  F TERM
000077          CALL SECORX (2)
000100          DO 140 I=1,N
000102    140   DELX(I)=PGRAD(I)
000106          CALL INVERX (1)
000107          DO 150 I=1,N
000111          X1(I)=DELX(I)
000113    150   DELX(I)=DELX0(I)
000116          CALL SECORX (2)
000120          CALL INVERX (1)
000122          DO 160 I=1,N
000124    160   XR2(I)=DELX(I)
000130          GO TO (170,200), NPAR2
000136    170   DOT1=0.
```

```
000137          DOT2=0.                                                 012120
000140          DO 180 I=1,N                                            012130
000141          DOT1=DOT1+PGRAD(I)*X1(I)                                012140
000144    180   DOT2=DOT2+DELXO(I)*XR2(I)                              012150
000151          RHO=SQRT(ABS(DOT1/DOT2))                                012160
000156          GO TO 20                                                012170
000156    190   NPAR2=2                                                 012180
          C     RHO MINIMIZES  2ND ORDER MOVE                           012190
000157          GO TO 120                                               012200
          C     USES  INTERNAL  SUB. TO COM  /DDP/-1 DF AND  /DDP/- DR  012210
000160    200   DOT1=0.0                                                012220
000161          DOT2=0.0                                                012230
000162          DO 210 I=1,N                                            012240
000163          DOT1=X1(I)**2+DOT1                                      012250
000166    210   DOT2=X1(I)*XR2(I)+DOT2                                  012260
000172          RHO=ABS(DOT1/DOT2)                                      012270
000175          GO TO 20                                                012280
000175          END                                                     012290
```

```
                SUBROUTINE SECORX(IS)
          C
          C                OCTOBER 1970
          C
          C SECORD EVALUATES THE MATRIX OF SECOND PARTIALS OF THE PENALTY
          C FUNCTION.
          C
          C- (1) MEANS DONT   COMPUTE GRAD. OUTER PRODUCT(IN SECORD)
000003          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000003          COMMON /EQALS/H, H1, MZ
000003          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000003          COMMON /VALUES/F,G,P0,RS1GMA, RJ(200), RHO
000003          COMMON/CRSTS/DELX(100), DELX0(100), RHOIN,RATIO, EPS1, THETA0,
               1 RS1G1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), DOTT, PGRAD(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMIN1, NPHASE, NSATIS
000003          DO 10 I=1,N
000005          DO 10 J=I,N
000006    10    A(I,J)=0.
000017          GO TO (230,20), IS
          CGRAD. TERM NOT PREV. COMPUTED
000024    20    DO 30 I=1,N
000026          DO 30 J=1,I
000027          A(I,J)=0.0
000033    30    CONTINUE
000037          GO TO (40,60), NT2
000045    40    DO 50 I=1,N
000047    50    A(I,I)=RHO/X(I)**2
000060    60    CONTINUE
000060          IF (M.LE.0) GO TO 130
000062          DO 120 IN=1,M
000063          IF (RJ(IN)) 120,120,70
000065    70    CALL GRADIS(IN)
000067          TT = RHO/RJ(IN)**2
000072          DO 110 I=1,N
000074          IF (DEL(I)) 80,110,80
000075    80    T=TT*DEL(I)
000100          DO 100 J=1,I
000101          IF (DEL(J)) 90,100,90
000102    90    A(I,J)=A(I,J)+T*DEL(J)
000110    100   CONTINUE
000113    110   CONTINUE
000116    120   CONTINUE
          C EQUALITY  CONSTRAINTS
000121    130   IF (MZ) 210,210,140
000123    140   GO TO (210,150,230), NPHASE
000132    150   RQ=2./RHO
000134          DO 200 JJ=1,MZ
000136          IN=M+JJ
000140          CALL GRADIS(IN)
000141          DO 190 I=1,N
000144          IF (DEL(I)) 160,190,160
000145    160   T=RQ*DEL(I)
000150          DO 180 J=1,I
000151          IF (DEL(J)) 170,180,170
000152    170   A(I,J)=A(I,J)+T*DEL(J)
000160    180   CONTINUE
```

```
000163   190   CONTINUE                                                  012870
000166   200   CONTINUE                                                  012880
000170   210   DO 220 I=1,N                                              012890
000172         DIAG(I)=A(I,I)                                            012900
000176   220   A(I,I)=0.                                                 012910
        C  READY NOW FOR MATRIX OF 2ND PARTIALS OF RESTRAINTS           012920
000203   230   GO TO (240,510,520), NT10                                 012930
000212   240   IF (M.LE.0) GO TO 340                                     012940
000214         DO 330 IN=1,M                                             012950
00021 5        LORN=2                                                    012960
        C CONSTRAINT ASSUMED NONLINEAR                                  012970
000216         CALL MATRXX (IN,LORN)
000220         IF (LORN.LT.2) GO TO 330                                  012990
000224         IF (RJ(IN) .GT. 0.0) GO TO 280                            013000
000227         DO 260 I=2,N                                              013010
00023 0        IM1=I-1                                                   013020
000232         DO 260 J=1,IM1                                            013030
000233         IF (A(J,I)) 250,260,250                                   013040
000236   250   A(I,J)=A(I,J)-A(J,I)                                      013050
000245         A(J,I)=0.                                                 013060
000247   260   CONTINUE                                                  013070
000254         DO 270 I=1,N                                              013080
000256         DIAG(I)=DIAG(I)-A(I,I)                                    013090
000262   270   A(I,I)=0.0                                                013100
000270         GO TO 330                                                 013110
000270   280   T=-RHO/RJ(IN)                                             013120
000273         DO 300 I=2,N                                              013130
000274         IM1=I-1                                                   013140
000276         DO 300 J=1,IM1                                            013150
000277         IF (A(J,I)) 290,300,290                                   013160
000302   290   A(I,J)=A(I,J)+T*A(J,I)                                    013170
000312         A(J,I)=0.                                                 013180
000315   300   CONTINUE                                                  013190
000322         DO 320 I=1,N                                              013200
000324         IF (A(I,I)) 310,320,310                                   013210
000327   310   DIAG(I)=DIAG(I)+T*A(I,I)                                  013220
000335         A(I,I)=0.                                                 013230
000340   320   CONTINUE                                                  013240
000343   330   CONTINUE                                                  013250
000346   340   CONTINUE                                                  013260
000346         GO TO (520,350,520), NPHASE                               013270
000355   350   IF (MZ.EQ.0) GO TO 420                                    013280
        C--     EQUALITY  SECOND PARTIALS HERE                          013290
000356         IF (NT10.GE.2) GO TO 420                                  013300
000361         DO 410 II=1,MZ                                            013310
000362         IN=M+II                                                   013320
000364         LORN=2                                                    013330
000365         CALL MATRXX (IN,LORN)
000367         IF (LORN.LT.2) GO TO 410                                  013350
000373         T=2.*RJ(IN)/RHO                                           013360
000376         DO 380 I=2,N                                              013370
000377         IM1=I-1                                                   013380
000401         DO 370 J=1,IM1                                            013390
000402         IF (A(J,I)) 360,370,360                                   013400
000405   360   A(I,J)=A(I,J)+T*A(J,I)                                    013410
000415         A(J,I)=0.0                                                013420
000420   370   CONTINUE                                                  013430
000423   380   CONTINUE                                                  013440
```

```
000425          DO 400 I=1,N                                                    0I3450
000427          IF (A(I,I)) 390,400,390                                         0I3460
000432    390   DIAG(I)=DIAG(I)+T*A(I,I)                                       0I3470
000440          A(I,I)=0.0                                                      0I3480
000443    400   CONTINUE                                                        0I3490
000446    410   CONTINUE                                                        0I3500
          C  GET MATRIX OF 2ND PARTIALS OF OBJECTIVE FUNCTION                  0I3510
000451    420   LLL=2                                                           0I3520
000452          CALL MATRXX( 0,LLL)
000454          IF (LLL.LT.5) GO TO 490                                         0I3540
000460          DO 440 I=2,N                                                    0I3550
000461          IMI=I-I                                                         0I3560
000463          DO 440 J=I,IMI                                                  0I3570
000464          IF (A(J,I)) 430,440,430                                         0I3580
000467    430   A(I,J)=A(I,J)+A(J,I)                                            0I3590
000476    440   A(J,I)=A(I,J)                                                   0I3600
000511          DO 470 I=I,N                                                    0I3610
000513          IF (A(I,I)) 450,460,450                                         0I3620
000516    450   A(I,I)=DIAG(I)+A(I,I)                                           0I3630
000522          GO TO 470                                                       0I3640
000523    460   A(I,I)=DIAG(I)                                                  0I3650
000527    470   CONTINUE                                                        0I3660
000532    480   RETURN                                                          0I3670
000533    490   DO 500 I=I,N                                                    0I3680
000535          A(I,I)=DIAG(I)                                                  0I3690
000541          DO 500 J=I,N                                                    0I3700
000542    500   A(I,J)=A(J,I)                                                   0I3710
000556          GO TO 480                                                       0I3720
000556    510   GO TO (520,350,350), NPHASE                                     0I3730
000565    520   DO 530 I=2,N                                                    0I3740
000567          IM1=I-1                                                         0I3750
000571          DO 530 J=1,IM1                                                  0I3760
000572    530   A(J,I)=A(I,J)                                                   0I3770
000606          DO 540 I=1,N                                                    0I3780
000607    540   A(I,I)=DIAG(I)                                                  0I3790
000616          GO TO 480                                                       0I3800
000617          END                                                             0I3810
```

```
                SUBROUTINE SETS(TMMAX)
          C
          C     FEBUARY 1971
          C
          C SET STORES THE TIME AT WHICH THE PROBLEM IS BEGUN
000003          COMMON /TSWS/NSWW
000003          COMMON /TMXS/TMO,EXT,EXT90
          C
          C  SECOND GIVES JOB CPU EXECUTION TIME IN 1/1000 OF A SECOND
          C
000003          CALL SECOND (TMO)
000005          EXT=TMMAX+TMO
000007          EXT90= TMO + 0.90*TMMAX
000012          NSWW=1
000013          RETURN
000014          END
```

```
                SUBROUTINE STORES
          C
          C              OCTOBER 1970
          C
          C STORE STORES THE VALUES OF THE CURRENT POINT AND THE ASSOCIATED
          C VALUES OF THE FUNCTIONS IN A TEMPORARY AREA.
000002          COMMON/SHARES/X(100), DEL(100), A(100,100),N,M, MN,NP1,NM1
000002          COMMON /EQALS/H, H1, MZ
000002          COMMON /VALUES/F,G,P0,RSIGMA, RJ(200), RHO
000002          COMMON/CRSTS/DELX(100), DELX0(100), RHOIN,RATIO, EPSI, THETA0,
               1 RSIG1, G1,    X1(100), X2(100), X3(100), XR2(100), XR1(100), PR1,
               2 PR2,P1, F1, RJ1(200), DOTT, PGRAD(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMINI, NPHASE, NSATIS
000002          DO 10 I=1,N
000004    10    X1(I)=X(I)
000010          MMZ=M+MZ
000012          DO 20 J=1,MMZ
000013    20    RJ1(J)=RJ(J)
000017          P1=P0
000020          F1=F
000022          G1=G
000023          RSIG1=RSIGMA
000025          H1=H
000026          RETURN
000027          END
```

```
                SUBROUTINE TCHFCX
          C
          C              FEBUARY 1971
          C
          C TCHECK CHECKS THE NUMBER OF SECONDS THAT HAVE ELAPSED SINCE THE START
          C OF THE PROBLEM. IF THE SOLUTION IS TAKING LONGER THAN 90 PER-CENT
          C OF THE ESTIMATED MAXIMUM TIME, A SWITCH IS SET TO GIVE MORE OUTPUT.
000002          COMMON /OPTNSS/NT1,NT2,NT3,NT4,NT5,NT6,NT7,NT8,NT9,NT10
000002          COMMON /TSWS/NSWW
000002          COMMON /TMXS/TMO,EXT,EXT90
000002          CALL SECOND (SECS)
000004          IF (SECS.LT.EXT90) RETURN
          C GETTING CLOSE TO EXCEEDING THE TIME LIMIT SET OUTPUT OPTION TO GIVE
          C MORE OUTPUT.
000007          NT3=2
000010          X=SECS - TMO
000012          WRITE(6,10) X
000020    10    FORMAT (6X,5HTIME=,F9.3,8H SECONDS  )
000020          IF(SECS .GT. EXT) NSWW=2
000024          CALL OUTPUX (1)
000026          RETURN
000027          END
```

```
                SUBROUTINE TIMECS
          C
          C     FEBUARY 1971
          C
          C TIMEC CHECKS THE NUMBER OF SECONDS THAT HAVE ELAPSED SINCE THE START
          C OF THE PROBLEM. IT PRINTS THIS NUMBER. IF THE SOLUTION IS TAKING
          C TIMEC CHECKS THE NUMBER OF SECONDS THAT HAVE ELAPSED SINCE THE START
          C OF THE PROBLEM. IT PRINTS THIS NUMBER. IF THE SOLUTION IS TAKING
          C LONGER THAN THE ESTIMATED MAXIMUM TIME, A SWITCH IS SET TO TERMINATE
          C THE RUN.
000002          COMMON /TSWS/NSWW
000002          COMMON /TMXS/TMO,EXT,EXT90
          C
          C  SECOND GIVES JOB CPU EXECUTION TIME IN 1/1000 OF A SECOND
          C
000002          CALL SECOND (SECS)
000004          X=SECS-TMO
000006          WRITE (6,20) X
000014          IF (SECS.LT.EXT) GO TO 10
000017          NSWW=2
000020    10    RETURN
          C
000021    20    FORMAT (6X,5HTIME=F9.3,8H SECONDS)
000021          END
```

```
                SUBROUTINE XMOVES
          C
          C                MARCH 1971
          C
          C XMOVE DETERMINES THE VECTOR ALONG WHICH THE SEARCH FOR A MINIMUM IS
          C USING OPT.
000002          COMMON/SHARES/X(100), DFL(100), A(100,100),N,M, MN,NP1,NM1
000002          COMMON/CRSTS/DFLX(100), DELX0(100), RHOIN,RATIO, EPS1, THETA0,
               1 RSIGI, GI,    X1(100), X2(100), X3(100), XR2(100), XRI(100), PR1,
               2 PR2,PI, FI, RJ1(200), DOTT, PGRAD(100), DIAG(100),
               3 PREV3,ADELX, NTCTR, NUMIN1, NPHASE, NSATIS
000002          COMMON/EXPDPX / NEXOP1, NEXOP2, XEP1, XEP2
000002          COMMON /XVFS/STG(100),YY(100),XXX(100),DELL(100)
          C--NEXOP2 DETERMINES HOW MOVE IS TO BE MADE
          C NEXOP2 = 1  USE MODIFIED  NEWTON RAPHSON METHOD.
          C        = 2  USE MODIFIED  NEWTON RAPHSON METHOD, BUT ADD DELX0 TO
          C             ORTHOGONAL MOVE VECTOR IF HESSIAN IS INDEFINITE.
          C        = 3  USE STEEPEST DESCENT METHOD.
          C        = 4  USE MCCORMICK'S MODIFICATION OF THE FLETCHER-POWELL
          C             METHOD.
000002          GO TO (10,10,180,30), NEXOP2
          C--NEWTON -RAPH WITH WHATEVER METHOD IS IN INVERSE
000012    10    CALL GRADS(1)
          C--ONE (I) MEANS ACCUMULATE MATRIX OF SECOND PARTIAL DERIVATIVES
000014          CALL SECORX (I)
000016          DO 20 I=1,N
000020    20    DELX(I)=DELX0(I)
000024          CALL INVERX(1)
          C IF A NONPOSITVE PIVOT IS ENCOUNTERED IN INVERSE AN ATTEMPT IS MADE TO
          C COMPUTE A VECTOR HAVING A POSITIVE DOT PRODUCT WITH A NEGATIVE
          C EIGENVECTOR AND THE NEGATIVE OF DEL P.
000025          CALL STORES
000026          CALL OPTS
000027          RETURN
          C--F-P-D-MCC MOVE
000030    30    CALL GRADS(2)
          C--MN IS NO. OF MOVES FOR THIS VALUE OF RHO
000032          IF (MN.NE.0) GO TO 70
000033    40    IREP=0
000034          IT=0
          C--SET INITIAL GUESS INVERS MATRIX OF SECOND PARTIAL DERIVATIVES
          C-- USE PARTIAL INVERSE IF KNOWN
000035          DO 50 I=1,N
000036          DO 50 J=I,N
000037    50    A(I,J)=0.0
000050          DO 60 I=1,N
000051    60    A(I,I)=1.0
000060    70    DO 80 I=1,N
000062    80    DELX(I)=DELX0(I)
000066          IF (IREP.GT.N) GO TO 40
000071          IF (IT.EQ.0) GO TO 130
000072          DO 90 I=1,N
000073          SIG(I)=X(I)-XXX(I)
000076    90    YY(I)=DELL(I)-DELX0(I)
          C--NEGATIVE GRADIENT STORED  AND COMPUTED
          C--COMPUTE  HY
000102          DO 100 I=1,N
```

```
000104          DELX(1)=0.0                                              015180
000105          DO 100 J=1,N                                             015190
000107    100   DELX(I)=DELX(1)+A(I,J)*YY(J)                             015200
          C--COMPUTE  Y(SIG -HY)-1                                       015210
000122          ZCON=0.0                                                 015220
000123          DO 110 1=1,N                                             015230
000124    110   ZCON=ZCON+YY(I)*(SIG(I)-DELX(1))                         015240
000132          IF (ZCON.EQ.0.0) GO TO 130                               015250
000133          IREP=IREP+1                                              015260
000135          ZC=1./ZCON                                               015270
          C-- UPDATE   H MATRIX USING  MCC FORMULA WHEN SCALAR NE 0      015280
000136          DO 120 1=1,N                                             015290
000140          T1=ZC*(SIG(1)-DELX(I))                                   015300
000143          DO 120 J=1,N                                             015310
000144          A(1,J)=A(1,J)+T1*(-DELX(J)+SIG(J))                       015320
000153    120   A(J,1)=A(I,J)                                            015330
          C-- STORE CURRENT POINT AND CURRENT GRADIENT (NEG)             015340
000165    130   DO 140 I=1,N                                             015350
000167          XXX(1)=X(1)                                              015360
000171    140   DELL(I)=DELX0(1)                                         015370
000174          DO 150 I=1,N                                             015380
000176          DELX(1)=0.0                                              015390
000177          DO 150 J=1,N                                             015400
000201    150   DELX(I)=DELX(1)+A(1,J)*DELX0(J)                          015410
000214          ZC1=0.0                                                  015420
000215          DO 160 I=1,N                                             015430
000216    160   ZC1=DELX(1)**2+ZC1                                       015440
000223          ZC1=SQRT(ZC1)                                            015450
000225          DO 170 1=1,N                                             015460
000226    170   DELX(1)=DELX(1)/ZC1                                      015470
000232          CALL STORES
000233          CALL OPTS
000234          IT=IT+1                                                  015500
000236          RETURN                                                   015510
000236    180   CONTINUE                                                 015520
          C STEEPEST DESCENT                                             015530
000236          CALL GRADS(E)
000240          DO 190 1=1,N                                             015550
000242    190   DELX(I)=DELX0(1)                                         015560
000246          CALL STORES
000247          CALL OPTS
000250          RETURN                                                   015590
000251          END                                                      015600
```

User-Supplied Subroutines for Inside Programs

```
                 SUBROUTINE READIX
000002     9999  CONTINUE
000002           RETURN
000003           END
```

```
                SUBROUTINE RESTNX(IN,VAL)
000005          COMMON/PROB/ISP
000005          COMMON/IN/W(100)
000005          COMMON/SHARES/X(100),DEL(100),A(100,100),N,M,MN,NP1,NM1
000005          IF(ISP-1)1000,1000,2000
000007     1000 FN=N
000011          VAL=0.
000011          IF(IN)1100,1100,1200
000013     1100 DO 1150 J=1,N
000015     1150 VAL=VAL+W(J)*(X(J)-2.)**2
000023          GO TO 9999
000024     1200 VAL=FN
000025          DO 1250 J=1,N
000027     1250 VAL=VAL-X(J)
000033          GO TO 9999
000033     2000 FN=N
000035          VAL=0.
000035          IF(IN)2100,2100,2200
000037     2100 DO 2150 J=1,N
000041     2150 VAL = VAL + W(J)**.5 * (X(J)-2.)**2
000053          GO TO 9999
000054     2200 VAL=FN
000055          DO 2250 J=1,N
000057     2250 VAL=VAL-X(J)**2
000063          GO TO 9999
000064     9999 CONTINUE
000064          RETURN
000065          END
```

```
                SUBROUTINE GRAD1S(IN)
000003          COMMON/PROB/ISP
000003          COMMON/IN/W(100)
000003          COMMON/SHARES/X(100),DEL(100),A(100,100),N,M,MN,NP1,NM1
000003          DO 500 J=1,N
000005      500 DEL(J)=0.
000010          IF(ISP-1)1000,1000,2000
000013     1000 IF(IN)1100,1100,1200
000015     1100 DO 1150 J=1,N
000017     1150 DEL(J) = 2.*W(J)*(X(J)-2.)
000025          GO TO 9999
000026     1200 DO 1250 J=1,N
000030     1250 DEL(J)=-1.
000034          GO TO 9999
000034     2000 IF(IN)2100,2100,2200
000036     2100 DO 2150 J=1,N
000040     2150 DEL(J) = 2. * W(J)**.5 * (X(J)-2.)
000052          GO TO 9999
000053     2200 DO 2250 J=1,N
000055     2250 DEL(J) =-2.*X(J)
000061          GO TO 9999
000062     9999 CONTINUE
000062          RETURN
000063          END
```

```
                SUBROUTINE MATRXX (IN,L)
000005          COMMON/PROB/ISP
000005          COMMON/IN/W(100)
000005          COMMON/SHARES/X(100),DEL(100),A(100,100),N,M,MN,NP1,NM1
000005          IF(ISP-1)1000,1000,2000
000007     1000 IF(IN)1100,1100,1200
000011     1100 DO 1150 J=1,N
000013     1150 A(J,J) = 2.*W(J)
000023          GO TO 9999
000023     1200 CONTINUE
000023          GO TO 9999
000024     2000 IF(IN)2100,2100,2200
000026     2100 DO 2150 J=1,N
000030     2150 A(J,J)= 2.*W(J)**.5
000044          GO TO 9999
000045     2200 DO 2250 J=1,N
000047     2250 A(J,J)=-2.
000056          GO TO 9999
000056     9999 CONTINUE
000056          RETURN
000057          END
```

USER-SUPPLIED INFORMATION CARDS AND RESULTS PRINTING OUT ALL POINTS OF INSIDE PROGRAMS

It is useful in debugging the overall program and in increasing solution efficiency to print out the solutions of the inside programs, preferably all points. The starting points of the inside programs should be feasible, which is dependent upon the starting point of the outside program. The starting point of the outside program should also be feasible. The solutions of the inside program for the starting point of the outside program should be examined. Also, several points of the outside program, and the corresponding inside program solutions, should be examined.

The SUB computer program, leaving in the statements for printing of points of the inside program, provides the opportunity to accomplish the above, when used with the user-supplied information cards given here.

User-supplied information cards are: parameter card, initial starting point, option card, tolerance card and second option card. Three sets of user-supplied information cards are necessary for the example problem.

The outside program has, on the parameter card, information that EPSI = 1.E-05, RHOIN = 10., THETAO = 1.E-05, RATIO = 10., TMMAX = 360., M = 2, N = 4, MZ = 0. The initial starting point is $x_1 = x_2 = x_3 = x_4 = 4$. The option card indicates that all points of the outside program are to be printed out (Option 3 = 2), and that there is at least one nonlinear constraint (Option 10 = 1). The tolerance card indicates that for numerical differentiation and control of P-function minimization the tolerances are .001. Finally, the second option card indicates that the problem is to be solved without checking first derivatives (Option 1 = 1), and that steepest descent is used to minimize the P-function (Option 2 = 3).

The inside program information cards are similar, with the following exceptions. Each problem has one constraint (M = 1). The starting points are $v_1 = v_2 = v_3 = v_4 = .5$. The first option cards indicate that all subproblem solutions of the inside programs are to be printed out (Option 3 = 1). The first option cards also indicate that the first inside program has all linear constraints (Option 10 = 2) and the second inside program has at least one non-linear constraint (Option 10 = 1). The second option cards for both inside programs indicate that the second-order method, called generalized Newton-Raphson, is used to minimize the P-function (Option 2 = 1).

The printout of results gives initial information for the outside program.

It indicates that Problem A is being solved. It gives initial information for Problem A, noting that SUB is used. It solves Problem A for the initial starting point of the outside program, requiring 16 points, and indicates that Problem A is solved.

It indicates that Problem B is being solved. It gives initial information for Problem B, noting that SUB is used. It solves Problem B for the initial starting point of the outside program, requiring 16 points, and indicates that Problem B is solved.

It prints out information for the outside program, namely the starting point values of the constraints found by solving Problem A and Problem B, and the feasible starting point to be used in the overall procedure.

It then prints that Problem A is being solved. The initial information is not printed this time, however, it having been by-passed in SUB after the first time. Again, Problem A is solved. The printout is given up to point 4 of this solution. The remainder, not included here, is similar for more points of the outside program.

User-Supplied Information Cards

```
1.E-05          10.        1.E-05          10.          360.   2   4   0
    4.            4.            4.            4.
3       1       2       1       1       1       1       1       1       1
  .001          .001
1       3
1.E-05          100.       1.E-05          10.          180.   1   4   0
    .5            .5            .5            .5
3       1       1       1       1       1       1       1       1       2
  .001          .001
1       1
1.E-05          100.       1.E-05          10.          180.   1   4   0
    .5            .5            .5            .5
3       1       1       1       1       1       1       1       1       1
  .001          .001
1       1
```

Printout of Results

```
NONLINEAR PROGRAMMING ROUTINE-SUMT VERSION 4  3/22/71
    N=  4       M=  2       MZ=  0
      MAX. TIME= 3.6000000E+02    R= 1.0000000E+01    RATIO= 1.0000000E+01      EPSILON= 1.0000000E-05    THETA= 1.0000000E-05
OPTIONS SELECTED
    3       1       2       1       1       1       1       1       1       1
TOLERANCES
     1.0000000E-03       1.0000000E-03
SECOND SET OF OPTIONS
    1       3
    TIME=     .025 SECONDS
```

PROBLEM A

```
NONLINEAR PROGRAMMING ROUTINE-SUMT VERSION 4     SUB

    N=  4      M=  1      MZ=  0

      MAX. TIME= 1.8000000E+02    R= 1.0000000E+02    RATIO= 1.0000000E+01      EPSILON= 1.0000000E-05    THETA= 1.0000000E-05

OPTIONS SELECTED
    3      1      1      1      1      1      1      1      1      2

TOLERANCES
     1.0000000E-03       1.0000000E-03

SECOND SET OF OPTIONS
    1      1
    TIME=    0.000 SECONDS
      F=  3.6000000E+01     P=  0.                G=  0.                RSIGMA=  0.                H=  0.
    THE CURRENT VALUE OF X IS
     5.0000000E-01       5.0000000E-01       5.0000000E-01       5.0000000E-01
    THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
     2.0000000E+00
    TIME=     .014 SECONDS

*****THE FEASIBLE STARTING POINT TO BE USED IS ...
      F=  3.6000000E+01     P=  0.                G=  0.                RSIGMA=  0.                H=  0.
    THE CURRENT VALUE OF X IS
     5.0000000E-01       5.0000000E-01       5.0000000E-01       5.0000000E-01
    THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
     2.0000000E+00
    TIME=     .073 SECONDS
*****************************************
        POINT=   2       DOTT=  6.4890716E-07      RHO=  1.0000000E+02      MAGNITUDE=  2.3681602E-02      PHASE= 2
      F=  2.2595486E+01     P=  1.3438514E+02     G= -4.7740451E+02     RSIGMA=  1.1178965E+02     H=  0.
    THE CURRENT VALUE OF X IS
     8.1163225E-01       8.1163225E-01       8.1163225E-01       8.1163225E-01
    THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
     7.5347099E-01


  LAGRANGE MULTIPLIERS
      F=  2.2595486E+01     P=  1.3438514E+02     G= -4.7740451E+02     RSIGMA=  1.1178965E+02     H=  0.
    THE CURRENT VALUE OF X IS
     1.2320851E+02       1.2320851E+02       1.2320851E+02       1.2320851E+02
    THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
     1.3271911E+02
    TIME=     .145 SECONDS
*****************************************
        POINT=   4       DOTT=  1.3778905E-07      RHO=  1.0000000E+01      MAGNITUDE=  5.0772586E-03      PHASE= 2
      F=  2.0166114E+01     P=  3.2520721E+01     G= -2.9833886E+01     RSIGMA=  1.2354607E+01     H=  0.
    THE CURRENT VALUE OF X IS
     8.7733258E-01       8.7733258E-01       8.7733258E-01       8.7733258E-01
    THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
     4.9066966E-01


   1ST  ORDER ESTIMATES
      F=  1.9904710E+01     P=  2.1202452E+01     G=  1.9896184E+01     RSIGMA=  0.                H=  0.
    THE CURRENT VALUE OF X IS
```

```
 8.8463262E-01        8.8463262E-01        8.8463262E-01        8.8463262E-01
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 4.6146952E-01


LAGRANGE MULTIPLIERS
   F=  1.9904710E+01     P=  2.1202452E+01     G=  1.9896184E+01     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 1.1398186E+01        1.1398186E+01        1.1398186E+01        1.1398186E+01
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 2.0380310E+01
TIME=      .233 SECONDS
*****************************************
      POINT=   6       DDTT=  3.6201352E-07      RHO=  1.0000000E+00      MAGNITUDE=  1.1263842E-02      PHASE= 2
   F=  1.6876951E+01     P=  1.9210771E+01     G=  1.1876951E+01     RSIGMA=  2.3338206E+00     H=  0.
THE CURRENT VALUE OF X IS
 9.7296084E-01        9.7296084E-01        9.7296084E-01        9.7296084E-01
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 1.0815663E-01


2ND  ORDER ESTIMATES
   F=  1.6497058E+01     P=  1.7696832E+01     G=  1.6511488E+01     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 9.8458573E-01        9.8458573E-01        9.8458573E-01        9.8458573E-01
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 6.1657062E-02


1ST  ORDER ESTIMATES
   F=  1.6529552E+01     P=  1.7731888E+01     G=  1.6511488E+01     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 9.8358620E-01        9.8358620E-01        9.8358620E-01        9.8358620E-01
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 6.5655187E-02


LAGRANGE MULTIPLIERS
   F=  1.6529552E+01     P=  1.7731888E+01     G=  1.6511488E+01     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 1.0277906E+00        1.0277906E+00        1.0277906E+00        1.0277906E+00
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 9.2458498E+00
TIME=      .335 SECONDS
*****************************************
      POINT=   8       DDTT=  5.5412877E-08      RHO=  1.0000000E-01      MAGNITUDE=  1.2123883E-02      PHASE= 2
   F=  1.6098637E+01     P=  1.6539599E+01     G=  1.5598637E+01     RSIGMA=  4.4096222E-01     H=  0.
THE CURRENT VALUE OF X IS
 9.9692234E-01        9.9692234E-01        9.9692234E-01        9.9692234E-01
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 1.2310625E-02


2ND  ORDER ESTIMATES
   F=  1.6008118E+01     P=  1.6227761E+01     G=  1.6012157E+01     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 9.9974633E-01        9.9974633E-01        9.9974633E-01        9.9974633E-01
```

```
THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
 1.0146632E-03

1ST  ORDER ESTIMATES
  F=  1.6013291E+01     P=  1.6242802E+01     G=  1.6012157E+01     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 9.9958473E-01       9.9958473E-01       9.9958473E-01       9.9958473E-01
THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
 1.6610684E-03

LAGRANGE MULTIPLIERS
  F=  1.6013291E+01     P=  1.6242802E+01     G=  1.6012157E+01     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 1.0030872E-01       1.0030872E-01       1.0030872E-01       1.0030872E-01
THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
 8.1230644E+00
TIME=     .433 SECONDS
*****************************************
    POINT=  10       DOTT=  5.8714668E-09      RHO=  1.0000000E-02      MAGNITUDE=  1.2290563E-02      PHASE= 2
  F=  1.6009988E+01     P=  1.6076860E+01     G=  1.5959988E+01     RSIGMA=  6.6871855E-02     H=  0.
THE CURRENT VALUE OF X IS
 9.9968791E-01       9.9968791E-01       9.9968791E-01       9.9968791E-01
THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
 1.2483446E-03

2ND  ORDER ESTIMATES
  F=  1.6000021E+01     P=  1.6023249E+01     G=  1.6000139E+01     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 9.9999935E-01       9.9999935E-01       9.9999935E-01       9.9999935E-01
THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
 2.6177795E-06

1ST  ORDER ESTIMATES
  F=  1.6000154E+01     P=  1.6025445E+01     G=  1.6000139E+01     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 9.9999520E-01       9.9999520E-01       9.9999520E-01       9.9999520E-01
THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
 1.9202286E-05

LAGRANGE MULTIPLIERS
  F=  1.6000154E+01     P=  1.6025445E+01     G=  1.6000139E+01     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 1.0003122E-02       1.0003122E-02       1.0003122E-02       1.0003122E-02
THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
 8.0106089E+00
TIME=     .527 SECONDS
*****************************************
    POINT=  12       DOTT=  1.9072340E-08      RHO=  1.0000000E-03      MAGNITUDE=  7.0193568E-02      PHASE= 2
  F=  1.6001001E+01     P=  1.6009987E+01     G=  1.5996001E+01     RSIGMA=  8.9861545E-03     H=  0.
THE CURRENT VALUE OF X IS
 9.9996871E-01       9.9996871E-01       9.9996871E-01       9.9996871E-01
THE CONSTRAINT VALUES
```

```
                    NOT INCLUDING THE NON-NEGATIVITIES
  1.2514601E-04

 2ND ORDER ESTIMATES
   F=  1.6000001E+01     P=  1.6002326E+01     G=  1.6000003E+01     RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   9.9999996E-01        9.9999996E-01        9.9999996E-01        9.9999996E-01
  THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
  1.5571086E-07

 1ST ORDER ESTIMATES
   F=  1.6000003E+01     P=  1.6002557E+01     G=  1.6000003E+01     RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   9.9999991E-01        9.9999991E-01        9.9999991E-01        9.9999991E-01
  THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
  3.4617658E-07

LAGRANGE MULTIPLIERS
   F=  1.6000003E+01     P=  1.6002557E+01     G=  1.6000003E+01     RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   1.0000313E-03        1.0000313E-03        1.0000313E-03        1.0000313E-03
  THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
  7.9906660E+00
  TIME=      .618 SECONDS
*****************************************
     POINT=  14        DOTT=  3.4340067E-07      RHO=  1.0000000E-04      MAGNITUDE=  8.8571925E-01      PHASE= 2
   F=  1.6000098E+01     P=  1.6001229E+01     G=  1.5999598E+01     RSIGMA=  1.1312673E-03     H=  0.
  THE CURRENT VALUE OF X IS
   9.9999695E-01        9.9999695E-01        9.9999695E-01        9.9999695E-01
  THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
  1.2217264E-05

 2ND ORDER ESTIMATES
   F=  1.5999997E+01     P=  1.6000233E+01     G=  1.5999997E+01     RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   1.0000001E+00        1.0000001E+00        1.0000001E+00        1.0000001E+00
  THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
  -3.3720913E-07

 1ST ORDER ESTIMATES
   F=  1.5999997E+01     P=  1.6000256E+01     G=  1.5999997E+01     RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   1.0000001E+00        1.0000001E+00        1.0000001E+00        1.0000001E+00
  THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
  -3.3037527E-07

LAGRANGE MULTIPLIERS
   F=  1.5999997E+01     P=  1.6000256E+01     G=  1.5999997E+01     RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   1.0000031E-04        1.0000031E-04        1.0000031E-04        1.0000031E-04
  THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
  8.1851389E+00
```

```
TIME=      .706 SECONDS
*****************************************
      POINT=  16        DOTT=  1.6627993E-07      RHO=  1.0000000E-05      MAGNITUDE=  1.8275368E+00      PHASE= 2
    F=  1.6000009E+01     P=  1.6000146E+01     G=  1.5999959E+01     RSIGMA=  1.3655964E-04     H=  0.
  THE CURRENT VALUE OF X IS
   9.9999971E-01       9.9999971E-01       9.9999971E-01       9.9999971E-01
  THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
   1.1729808E-06


 2ND  ORDER ESTIMATES
    F=  1.6000000E+01     P=  1.6000023E+01     G=  1.6000000E+01     RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   1.0000000E+00       1.0000000E+00       1.0000000E+00       1.0000000E+00
  THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
  -5.1371686E-08


 1ST  ORDER ESTIMATES
    F=  1.6000000E+01     P=  1.6000026E+01     G=  1.6000000E+01     RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   1.0000000E+00       1.0000000E+00       1.0000000E+00       1.0000000E+00
  THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
  -5.4161688E-08


LAGRANGE MULTIPLIERS
    F=  1.6000000E+01     P=  1.6000026E+01     G=  1.6000000E+01     RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   1.0000003E-05       1.0000003E-05       1.0000003E-05       1.0000003E-05
  THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
   8.5252888E+00
```

PROBLEM A SOLVED

PROBLEM H

```
NONLINEAR PROGRAMMING ROUTINE-SUMT VERSION 4     SUB

    N=  4       M=  1       MZ=  0

      MAX. TIME= 1.8000000E+02    R= 1.0000000E+02    RATIO= 1.0000000E+01      EPSILON= 1.0000000E-05    THETA= 1.0000000E-05

OPTIONS SELECTED
    3      1      1      1      1      1      1      1      1      1

TOLERANCES
     1.0000000E-03       1.0000000E-03

SECOND SET OF OPTIONS
    1      1
    TIME=    0.000 SECONDS
      F=  1.8000000E+01     P=  0.                G=  0.                RSIGMA=  0.                H=  0.
    THE CURRENT VALUE OF X IS
     5.0000000E-01       5.0000000E-01       5.0000000E-01       5.0000000E-01
    THE CONSTRAINT VALUES
                              NOT INCLUDING THE NON-NEGATIVITIES
     3.0000000E+00
    TIME=     .014 SECONDS

*****THE FEASIBLE STARTING POINT TO BE USED IS ...
      F=  1.8000000E+01     P=  0.                G=  0.                RSIGMA=  0.                H=  0.
    THE CURRENT VALUE OF X IS
     5.0000000E-01       5.0000000E-01       5.0000000E-01       5.0000000E-01
    THE CONSTRAINT VALUES
                              NOT INCLUDING THE NON-NEGATIVITIES
     3.0000000E+00
    TIME=     .112 SECONDS

*****************************************
         POINT=   2        DOTT=  6.6344653E-07      RHO=  1.0000000E+02      MAGNITUDE=  2.5042075E-02      PHASE= 2
      F=  1.1108596E+01     P=  6.3481406E+01     G= -4.8889140E+02     RSIGMA=  5.2372810E+01     H=  0.
    THE CURRENT VALUE OF X IS
     8.2162208E-01       8.2162208E-01       8.2162208E-01       8.2162208E-01
    THE CONSTRAINT VALUES
                              NOT INCLUDING THE NON-NEGATIVITIES
     1.2997486E+00


 LAGRANGE MULTIPLIERS
      F=  1.1108596E+01     P=  6.3481406E+01     G= -4.8889140E+02     RSIGMA=  5.2372810E+01     H=  0.
    THE CURRENT VALUE OF X IS
     1.2171046E+02       1.2171046E+02       1.2171046E+02       1.2171046E+02
    THE CONSTRAINT VALUES
                              NOT INCLUDING THE NON-NEGATIVITIES
     7.6937953E+01
    TIME=     .217 SECONDS

*****************************************
         POINT=   4        DOTT=  7.4900159E-08      RHO=  1.0000000E+01      MAGNITUDE=  3.2592589E-03      PHASE= 2
      F=  1.0438018E+01     P=  1.6020672E+01     G= -3.9561982E+01     RSIGMA=  5.5826542E+00     H=  0.
    THE CURRENT VALUE OF X IS
     8.5774248E-01       8.5774248E-01       8.5774248E-01       8.5774248E-01
    THE CONSTRAINT VALUES
                              NOT INCLUDING THE NON-NEGATIVITIES
     1.0571113E+00


   1ST  ORDER ESTIMATES
      F=  1.0364798E+01     P=  1.0747257E+01     G=  1.0363509E+01     RSIGMA=  0.                H=  0.
    THE CURRENT VALUE OF X IS
```

```
   8.6175586E-01        8.6175586E-01        8.6175586E-01        8.6175586E-01
  THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
   1.0295073E+00


LAGRANGE MULTIPLIERS
    F=  1.0364798E+01     P=  1.0747257E+01     G=  1.0363509E+01     RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   1.1658511E+01        1.1658511E+01        1.1658511E+01        1.1658511E+01
  THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
   9.4597417E+00
  TIME=      .341 SECONDS
*****************************************
      POINT=   6        DOTT=  3.4565497E-10      RHO=  1.0000000E+00      MAGNITUDE=  2.0385174E-04      PHASE= 2
    F=  8.7631835E+00     P=  9.9641443E+00     G=  3.7631835E+00     RSIGMA=  1.2009608E+00     H=  0.
  THE CURRENT VALUE OF X IS
   9.5338740E-01        9.5338740E-01        9.5338740E-01        9.5338740E-01
  THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
   3.6420989E-01


 2ND  ORDER ESTIMATES
    F=  8.5690130E+00     P=  9.2764891E+00     G=  8.5770908E+00     RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   9.6504753E-01        9.6504753E-01        9.6504753E-01        9.6504753E-01
  THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
   2.7473309E-01


 1ST  ORDER ESTIMATES
    F=  8.5861258E+00     P=  9.2911968E+00     G=  8.5770908E+00     RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   9.6401461E-01        9.6401461E-01        9.6401461E-01        9.6401461E-01
  THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
   2.8270333E-01


LAGRANGE MULTIPLIERS
    F=  8.5861258E+00     P=  9.2911968E+00     G=  8.5770908E+00     RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   1.0488916E+00        1.0488916E+00        1.0488916E+00        1.0488916E+00
  THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
   2.7456695E+00
  TIME=      .480 SECONDS
*****************************************
      POINT=   8        DOTT=  1.9350286E-08      RHO=  1.0000000E-01      MAGNITUDE=  3.6524186E-03      PHASE= 2
    F=  8.0969863E+00     P=  8.4026486E+00     G=  7.5969863E+00     RSIGMA=  3.0566231E-01     H=  0.
  THE CURRENT VALUE OF X IS
   9.9395662E-01        9.9395662E-01        9.9395662E-01        9.9395662E-01
  THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
   4.8200967E-02


 2ND  ORDER ESTIMATES
    F=  8.0190147E+00     P=  8.2184213E+00     G=  8.0229644E+00     RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   9.9881229E-01        9.9881229E-01        9.9881229E-01        9.9881229E-01
```

```
THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
 9.4960700E-03


IST  ORDER ESTIMATES
  F=  8.0245899E+00     P=  8.2291491E+00     G=  8.0229644E+00     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 9.9846431E-01       9.9846431E-01       9.9846431E-01       9.9846431E-01
THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
 1.2276093E-02


LAGRANGE MULTIPLIERS
  F=  8.0245899E+00     P=  8.2291491E+00     G=  8.0229644E+00     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 1.0060801E-01       1.0060801E-01       1.0060801E-01       1.0060801E-01
THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
 2.0746472E+00
TIME=     .609 SECONDS
*****************************************
    POINT=  10       OOTT=  8.6014629E-09      RHO=  1.0000000E-02      MAGNITUDE=  7.4404491E-03      PHASE= 2
  F=  8.0099660E+00     P=  8.0630144E+00     G=  7.9599660E+00     RSIGMA=  5.3048390E-02     H=  0.
THE CURRENT VALUE OF X IS
 9.9937732E-01       9.9937732E-01       9.9937732E-01       9.9937732E-01
THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
 4.9798896E-03


2ND  ORDER ESTIMATES
  F=  8.0000812E+00     P=  8.0232179E+00     G=  8.0002971E+00     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 9.9999493E-01       9.9999493E-01       9.9999493E-01       9.9999493E-01
THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
 4.0589271E-05


IST  ORDER ESTIMATES
  F=  8.0003261E+00     P=  8.0252772E+00     G=  8.0002971E+00     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 9.9997962E-01       9.9997962E-01       9.9997962E-01       9.9997962E-01
THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
 1.6303709E-04


LAGRANGE MULTIPLIERS
  F=  8.0003261E+00     P=  8.0252772E+00     G=  8.0002971E+00     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 1.0006231E-02       1.0006231E-02       1.0006231E-02       1.0006231E-02
THE CONSTRAINT VALUES
                    NOT INCLUDING THE NON-NEGATIVITIES
 2.0080766E+00
TIME=     .738 SECONDS
*****************************************
    POINT=  12       OOTT=  5.6830703E-08      RHO=  1.0000000E-03      MAGNITUDE=  5.9880157E-02      PHASE= 2
  F=  8.0009973E+00     P=  8.0086012E+00     G=  7.9959973E+00     RSIGMA=  7.6039334E-03     H=  0.
THE CURRENT VALUE OF X IS
 9.9993767E-01       9.9993767E-01       9.9993767E-01       9.9993767E-01
THE CONSTRAINT VALUES
```

```
                         NOT INCLUDING THE NON-NEGATIVITIES
   4.9861113E-04

  2ND  ORDER ESTIMATES
    F=  7.9999978E+00     P=  8.0023258E+00     G=  8.0000008E+00     RSIGMA=  0.                H=  0.
   THE CURRENT VALUE OF X IS
    1.0000001E+00       1.0000001E+00       1.0000001E+00       1.0000001E+00
   THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
   -1.1052083E-06

  1ST  ORDER ESTIMATES
    F=  8.0000011E+00     P=  8.0025553E+00     G=  8.0000008E+00     RSIGMA=  0.                H=  0.
   THE CURRENT VALUE OF X IS
    9.9999993E-01       9.9999993E-01       9.9999993E-01       9.9999993E-01
   THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
   5.3623128E-07

 LAGRANGE MULTIPLIERS
    F=  8.0000011E+00     P=  8.0025553E+00     G=  8.0000008E+00     RSIGMA=  0.                H=  0.
   THE CURRENT VALUE OF X IS
    1.0000623E-03       1.0000623E-03       1.0000623E-03       1.0000623E-03
   THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
   2.0055710E+00
   TIME=     .856 SECONDS
*****************************************
      POINT=  14        DOTT=  1.8815766E-07      RHO=  1.0000000E-04      MAGNITUDE=  3.6276577E-01      PHASE= 2
    F=  8.0001011E+00     P=  8.0010904E+00     G=  7.9996011E+00     RSIGMA=  9.8926685E-04     H=  0.
   THE CURRENT VALUE OF X IS
    9.9999368E-01       9.9999368E-01       9.9999368E-01       9.9999368E-01
   THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
   5.0545165E-05

  2ND  ORDER ESTIMATES
    F=  8.0000015E+00     P=  8.0002326E+00     G=  8.0000015E+00     RSIGMA=  0.                H=  0.
   THE CURRENT VALUE OF X IS
    9.9999990E-01       9.9999990E-01       9.9999990E-01       9.9999990E-01
   THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
   7.6075462E-07

  1ST  ORDER ESTIMATES
    F=  8.0000015E+00     P=  8.0002558E+00     G=  8.0000015E+00     RSIGMA=  0.                H=  0.
   THE CURRENT VALUE OF X IS
    9.9999991E-01       9.9999991E-01       9.9999991E-01       9.9999991E-01
   THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
   7.5850938E-07

 LAGRANGE MULTIPLIERS
    F=  8.0000015E+00     P=  8.0002558E+00     G=  8.0000015E+00     RSIGMA=  0.                H=  0.
   THE CURRENT VALUE OF X IS
    1.0000063E-04       1.0000063E-04       1.0000063E-04       1.0000063E-04
   THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
   1.9784286E+00
```

```
TIME=      .969 SECONDS
*****************************************
      POINT=  16        DOTT=  4.0246536E-07      RHO=  1.0000000E-05      MAGNITUDE=  1.336755TE+00      PHASE= 2
    F=  8.0000089E+00      P=  8.000132IE+00      G=  7.9999589E+00      RSIGMA=  I.232T88IE-04      H=  0.
  THE CURRENT VALUE OF X IS
   9.9999945E-0I        7.9999945E-01        9.9999945E-01        9.9999945E-0I
  THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
   4.4265993E-06


 2ND  ORDER ESTIMATES
    F=  T.9999986E+00      P=  8.0000233E+00      G=  T.9999986E+00      RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   1.0000001E+00        1.0000001E+00        I.000000IE+00        1.000000IE+00
  THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
  -T.I241I36E-07


 IST  ORDER ESTIMATES
    F=  T.9999986E+00      P=  8.000025TE+00      G=  T.9999986E+00      RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   I.0000001E+00        1.000000IE+00        1.0000001E+00        I.000000IE+00
  THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
  -6.9770213E-0T


LAGRANGE MULTIPLIERS
    F=  7.9999986E+00      P=  8.000025TE+00      G=  T.9999986E+00      RSIGMA=  0.                H=  0.
  THE CURRENT VALUE OF X IS
   1.0000006E-05        I.0000006E-05        1.0000006E-05        1.0000006E-05
  THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
   2.2590705E+00
```

```
PROBLEM 8 SOLVED
      F=  1.6000000E+01      P=  0.                G=  0.                RSIGMA=  0.                H=  0.
    THE CURRENT VALUE OF X IS
     4.0000000E+00        4.0000000E+00        4.0000000E+00        4.0000000E+00
    THE CONSTRAINT VALUES
                           NOT INCLUDING THE NON-NEGATIVITIES
     1.2000009E+01        4.0000089E+00
    TIME=     1.900 SECONDS

*****THE FEASIBLE STARTING POINT TO BE USED IS ...
      F=  1.6000000E+01      P=  0.                G=  0.                RSIGMA=  0.                H=  0.
    THE CURRENT VALUE OF X IS
     4.0000000E+00        4.0000000E+00        4.0000000E+00        4.0000000E+00
    THE CONSTRAINT VALUES
                           NOT INCLUDING THE NON-NEGATIVITIES
     1.2000009E+01        4.0000089E+00
```

```
PROBLEM A
   TIME=    0.000 SECONDS
     F=  3.6000000E+01      P=  0.                G=  0.                SIGMA=  0.                H=  0.
   THE CURRENT VALUE OF X IS
    5.0000000E-01        5.0000000E-01        5.0000000E-01        5.0000000E-01
   THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
    2.0000000E+00
   TIME=     .013 SECONDS

*****THE FEASIBLE STARTING POINT TO BE USED IS ...
     F=  3.6000000E+01      P=  0.                G=  0.                SIGMA=  0.                H=  0.
   THE CURRENT VALUE OF X IS
    5.0000000E-01        5.0000000E-01        5.0000000E-01        5.0000000E-01
   THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
    2.0000000E+00
   TIME=     .073 SECONDS

*****************************************
       POINT=   2        DOTT=  6.4890716E-07      RHO=  1.0000000E+02      MAGNITUDE=  2.3681602E-02      PHASE= 2
     F=  2.2595486E+01      P=  1.3438514E+02      G= -4.7740451E+02      SIGMA=  1.1178965E+02      H=  0.
   THE CURRENT VALUE OF X IS
    8.1163225E-01        8.1163225E-01        8.1163225E-01        8.1163225E-01
   THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
    7.5347099E-01


 LAGRANGE MULTIPLIERS
     F=  2.2595486E+01      P=  1.3438514E+02      G= -4.7740451E+02      SIGMA=  1.1178965E+02      H=  0.
   THE CURRENT VALUE OF X IS
    1.2320851E+02        1.2320851E+02        1.2320851E+02        1.2320851E+02
   THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
    1.3271911E+02
   TIME=     .145 SECONDS

*****************************************
       POINT=   4        DOTT=  1.3778905E-07      RHO=  1.0000000E+01      MAGNITUDE=  5.0772586E-03      PHASE= 2
     F=  2.0166114E+01      P=  3.2520721E+01      G= -2.9833886E+01      SIGMA=  1.2354607E+01      H=  0.
   THE CURRENT VALUE OF X IS
    8.7733258E-01        8.7733258E-01        8.7733258E-01        8.7733258E-01
   THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
    4.9066966E-01


  1ST  ORDER ESTIMATES
     F=  1.9904710E+01      P=  2.1202452E+01      G=  1.9896184E+01      SIGMA=  0.                H=  0.
   THE CURRENT VALUE OF X IS
    8.8463262E-01        8.8463262E-01        8.8463262E-01        8.8463262E-01
   THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
    4.6146952E-01


 LAGRANGE MULTIPLIERS
     F=  1.9904710E+01      P=  2.1202452E+01      G=  1.9896184E+01      SIGMA=  0.                H=  0.
   THE CURRENT VALUE OF X IS
    1.1398186E+01        1.1398186E+01        1.1398186E+01        1.1398186E+01
   THE CONSTRAINT VALUES
                         NOT INCLUDING THE NON-NEGATIVITIES
    2.0380310E+01
```

RESULTS PRINTING OUT ALL POINTS OF OUTSIDE PROGRAM

The SUMT program is modified by changing RESTNT, and the INSUMT program is modified by changing BODYS, CONVRX, ESTIMS, FEASS, INVERX, OPTS, OUTPUX, PUNCHS, TCHECX, and TIMECS to eliminate printing of inside programs.

The user-supplied information cards are identical to those of the previous section, and are not repeated here. The printouts of the inside programs called for by their information cards are by-passed by the changes in subroutines mentioned above.

The printout of the solution prints initial information for the outside program, for Problem A, and for Problem B. From then on all of the printout pertains to the outside program. At point 17 the value of r has been reduced to 1.E-04, and the solution has been approximately attained. The program cuts off at central processor time equals 360 seconds.

Printout of Results

```
NONLINEAR PROGRAMMING ROUTINE-SUMT VERSION 4  3/22/71
    N=  4       M=  2       MZ=  0
     MAX. TIME= 3.6000000E+02    R= 1.0000000E+01    RATIO= 1.0000000E+01      EPSILON= 1.0000000E-05    THETA= 1.0000000E-05
OPTIONS SELECTED
    3       1       2       1       1       1       1       1       1       1
TOLERANCES
     1.0000000E-03       1.0000000E-03
SECOND SET OF OPTIONS
    1       3
    TIME=      .027 SECONDS
```

```
NONLINEAR PROGRAMMING ROUTINE-SUMT VERSION 4     SUR
    N=  4       M=  1       MZ=  0

     MAX. TIME= 1.8000000E+02    R= 1.0000000E+02    RATIO= 1.0000000E+01      EPSILON= 1.0000000E-05    THETA= 1.0000000E-05
OPTIONS SELECTED
    3       1       1       1       1       1       1       1       1       2
TOLERANCES
     1.0000000E-03       1.0000000E-03
SECOND SET OF OPTIONS
    1       1
```

```
NONLINEAR PROGRAMMING ROUTINE-SUMT VERSION 4     SUB

    N=  4       M=  1       MZ=  0

      MAX. TIME= 1.8000000E+02    R= 1.0000000E+02    RATIO= 1.0000000E+01      EPSILON= 1.0000000E-05    THETA= 1.0000000E-05

OPTIONS SELECTED
    3      1      1      1      1      1      1      1      1      1

TOLERANCES
     1.0000000E-03        1.0000000E-03

SECOND SET OF OPTIONS
    1      1
      F=  1.6000000E+01      P=  0.                G=  0.                RSIGMA=  0.                H=  0.
    THE CURRENT VALUE OF X IS
     4.0000000E+00        4.0000000E+00        4.0000000E+00        4.0000000E+00
    THE CONSTRAINT VALUES
                          NOT INCLUDING THE NON-NEGATIVITIES
     1.2000099E+01        4.0000089E+00
    TIME=      .963 SECONDS

*****THE FEASIBLE STARTING POINT TO BE USED IS ...
      F=  1.6000000E+01      P=  0.                G=  0.                RSIGMA=  0.                H=  0.
    THE CURRENT VALUE OF X IS
     4.0000000E+00        4.0000000E+00        4.0000000E+00        4.0000000E+00
    THE CONSTRAINT VALUES
                          NOT INCLUDING THE NON-NEGATIVITIES
     1.2000099E+01        4.0000089E+00
    TIME=   23.600 SECONDS

*****************************************
        POINT=   1        DOTT=  3.5011227E+01      RHO=  1.0000000E+01      MAGNITUDE=  5.9170286E+00      PHASE= 2
      F=  5.7545151E+01      P= -1.1304499E+02      G= -2.4548487E+00      RSIGMA= -1.7059014E+02      H=  0.
    THE CURRENT VALUE OF X IS
     1.4386288E+01        1.4386288E+01        1.4386288E+01        1.4386288E+01
    THE CONSTRAINT VALUES
                          NOT INCLUDING THE NON-NEGATIVITIES
     5.3545259E+01        1.1171714E+01
    TIME=   35.111 SECONDS

*****************************************
        POINT=   2        DOTT=  5.1208486E-08      RHO=  1.0000000E+01      MAGNITUDE=  2.2629292E-04      PHASE= 2
      F=  5.7542415E+01      P= -1.1304499E+02      G= -2.4575846E+00      RSIGMA= -1.7058741E+02      H=  0.
    THE CURRENT VALUE OF X IS
     1.4385604E+01        1.4385604E+01        1.4385604E+01        1.4385604E+01
    THE CONSTRAINT VALUES
                          NOT INCLUDING THE NON-NEGATIVITIES
     5.3542518E+01        1.1171354E+01


  LAGRANGE MULTIPLIERS
      F=  5.7542415E+01      P= -1.1304499E+02      G= -2.4575846E+00      RSIGMA= -1.7058741E+02      H=  0.
    THE CURRENT VALUE OF X IS
     6.9513940E-01        6.9513940E-01        6.9513940E-01        6.9513940E-01
    THE CONSTRAINT VALUES
                          NOT INCLUDING THE NON-NEGATIVITIES
     1.8676747E-01        8.9514667E-01
    TIME=   64.810 SECONDS

*****************************************
        POINT=   3        DOTT=  3.2394380E+00      RHO=  1.0000000E+00      MAGNITUDE=  1.7998439E+00      PHASE= 2
      F=  7.8079487E+00      P=  3.3326411E+00      G=  1.8079487E+00      RSIGMA= -4.4753076E+00      H=  0.
    THE CURRENT VALUE OF X IS
```

```
 1.9519872E+00        1.9519872E+00        1.9519872E+00        1.9519872E+00
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 3.8079588E+00        1.5885151E+00
TIME=   80.232 SECONDS

*****************************************
      POINT=   5        DOTT=  2.6754227E-07      RHO=  1.0000000E+00      MAGNITUDE=  5.1724488E-04      PHASE= 2
  F=  7.8084659E+00      P=  3.3326410E+00      G=  1.8084659E+00      RSIGMA= -4.4758249E+00      H=  0.
THE CURRENT VALUE OF X IS
 1.9521165E+00        1.9521165E+00        1.9521165E+00        1.9521165E+00
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 3.8084760E+00        1.5887366E+00

1ST ORDER ESTIMATES
  F=  2.2824715E+00      P=  1.6263489E+01      G=  2.2824715E+00      RSIGMA=  0.                  H=  0.
THE CURRENT VALUE OF X IS
 5.7061789E-01        5.7061787E-01        5.7061787E-01        5.7061788E-01
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
-1.7175275E+00       -9.7842900E-01

LAGRANGE MULTIPLIERS
  F=  2.2824715E+00      P=  1.6263489E+01      G=  2.2824715E+00      RSIGMA=  0.                  H=  0.
THE CURRENT VALUE OF X IS
 5.1226451E-01        5.1226452E-01        5.1226452E-01        5.1226452E-01
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 2.6257222E-01        6.2943094E-01
TIME=  107.247 SECONDS

*****************************************
      POINT=   6        DOTT=  3.2399114E+00      RHO=  1.0000000E-01      MAGNITUDE=  1.7999754E+00      PHASE= 2
  F=  4.2193837E+00      P=  4.5720697E+00      G=  3.6193837E+00      RSIGMA=  3.5268605E-01      H=  0.
THE CURRENT VALUE OF X IS
 1.0548459E+00        1.0548459E+00        1.0548459E+00        1.0548459E+00
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 2.1938451E-01        1.0822856E-01
TIME=  124.671 SECONDS

*****************************************
      POINT=   8        DOTT=  7.6208011E-07      RHO=  1.0000000E-01      MAGNITUDE=  8.7297200E-04      PHASE= 2
  F=  4.2191776E+00      P=  4.5720699E+00      G=  3.6191776E+00      RSIGMA=  3.5289232E-01      H=  0.
THE CURRENT VALUE OF X IS
 1.0547944E+00        1.0547944E+00        1.0547944E+00        1.0547944E+00
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 2.1917841E-01        1.0812823E-01

2ND ORDER ESTIMATES
  F=  3.8359021E+00      P=  4.5930801E+00      G=  3.8203678E+00      RSIGMA=  0.                  H=  0.
THE CURRENT VALUE OF X IS
 9.5897553E-01        9.5897552E-01        9.5897552E-01        9.5897552E-01
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
-1.6409711E-01       -8.2907059E-02

1ST ORDER ESTIMATES
  F=  3.8203678E+00      P=  4.7097862E+00      G=  3.8203678E+00      RSIGMA=  0.                  H=  0.
THE CURRENT VALUE OF X IS
```

```
 9.5509196E-01        9.5509194E-01        9.5509194E-01        9.5509194E-01
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
-1.7963133E-01       -9.0846653E-02


LAGRANGE MULTIPLIERS
  F=  3.8203678E+00     P=  4.7097842E+00     G=  3.8203678E+00     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 9.4805205E-02        9.4805206E-02        9.4805206E-02        9.4805206E-02
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 4.5624932E-01        9.2482785E-01
TIME=  152.061 SECONDS
*****************************************
     POINT=   9        DOTT=  3.2390600E+00      RHO=  1.0000000E-02      MAGNITUDE=  1.7997389E+00      PHASE= 2
  F=  4.0201711E+00     P=  4.1049827E+00     G=  3.9601711E+00     RSIGMA=  8.4811687E-02     H=  0.
THE CURRENT VALUE OF X IS
 1.0050428E+00        1.0050428E+00        1.0050428E+00        1.0050428E+00
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 2.0171962E-02        1.0073698E-02
TIME=  176.878 SECONDS
*****************************************
     POINT=  11        DOTT=  2.6602785E-06      RHO=  1.0000000E-02      MAGNITUDE=  1.6310360E-03      PHASE= 2
  F=  4.0201149E+00     P=  4.1049829E+00     G=  3.9601149E+00     RSIGMA=  8.4867987E-02     H=  0.
THE CURRENT VALUE OF X IS
 1.0050287E+00        1.0050287E+00        1.0050287E+00        1.0050287E+00
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 2.0115784E-02        1.0045678E-02


2ND  ORDER ESTIMATES
  F=  3.9997910E+00     P=  4.0464510E+00     G=  3.9979968E+00     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 9.9994777E-01        9.9994776E-01        9.9994776E-01        9.9994776E-01
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
-2.0810406E-04       -1.0335564E-04


1ST  ORDER ESTIMATES
  F=  3.9979968E+00     P=  4.0530843E+00     G=  3.9979968E+00     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 9.9949921E-01        9.9949920E-01        9.9949920E-01        9.9949920E-01
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
-2.0022856E-03       -1.0008652E-03


LAGRANGE MULTIPLIERS
  F=  3.9979968E+00     P=  4.0530843E+00     G=  3.9979968E+00     RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 9.9499643E-03        9.9499644E-03        9.9499644E-03        9.9499644E-03
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 4.9712205E-01        9.9545294E-01
TIME=  201.330 SECONDS
*****************************************
     POINT=  12        DOTT=  3.2374195E+00      RHO=  1.0000000E-03      MAGNITUDE=  1.7992831E+00      PHASE= 2
  F=  4.0020147E+00     P=  4.0151190E+00     G=  3.9960147E+00     RSIGMA=  1.3104221E-02     H=  0.
THE CURRENT VALUE OF X IS
```

```
 1.0005037E+00        1.0005037E+00        1.0005037E+00        1.0005037E+00
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 2.0156495E-03        1.0083689E-03
TIME=  234.886 SECONDS

*****************************************
     POINT=  14        DOTT=  2.0009282E-04      RHO=  1.0000000E-03      MAGNITUDE=  1.4145417E-02      PHASE= 2
  F=  4.0020147E+00      P=  4.0151190E+00      G=  3.9960147E+00      RSIGMA=  1.3104221E-02      H=  0.
THE CURRENT VALUE OF X IS
 1.0005037E+00        1.0005037E+00        1.0005037E+00        1.0005037E+00
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 2.0156495E-03        1.0083689E-03
APPARENTLY ROUNDOFF ERRORS PREVENT A MORE ACCURATE DETERMINATION OF THE MINIMUM OF THIS SUBPROBLEM.

2ND  ORDER ESTIMATES
  F=  4.0000239E+00      P=  4.0046497E+00      G=  4.0000036E+00      RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 1.0000060E+00        1.0000060E+00        1.0000060E+00        1.0000060E+00
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 2.4754009E-05        1.3068135E-05

1ST  ORDER ESTIMATES
  F=  4.0000036E+00      P=  4.0051341E+00      G=  4.0000036E+00      RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 1.0000009E+00        1.0000009E+00        1.0000009E+00        1.0000009E+00
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 4.4821830E-06        2.9327844E-06

LAGRANGE MULTIPLIERS
  F=  4.0000036E+00      P=  4.0051341E+00      G=  4.0000036E+00      RSIGMA=  0.                H=  0.
THE CURRENT VALUE OF X IS
 9.9949656E-04        9.9949657E-04        9.9949657E-04        9.9949657E-04
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 4.9611801E-01        9.9170060E-01
TIME=  259.399 SECONDS

*****************************************
     POINT=  15        DOTT=  3.2450944E+00      RHO=  1.0000000E-04      MAGNITUDE=  1.8014145E+00      PHASE= 2
  F=  4.0001962E+00      P=  4.0019712E+00      G=  3.9995962E+00      RSIGMA=  1.7749474E-03      H=  0.
THE CURRENT VALUE OF X IS
 1.0000491E+00        1.0000491E+00        1.0000491E+00        1.0000491E+00
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 1.9711422E-04        9.9242635E-05
TIME=  294.894 SECONDS

*****************************************
     POINT=  17        DOTT=  5.0298191E-04      RHO=  1.0000000E-04      MAGNITUDE=  2.2427258E-02      PHASE= 2
  F=  4.0001900E+00      P=  4.0019714E+00      G=  3.9995900E+00      RSIGMA=  1.7814014E-03      H=  0.
THE CURRENT VALUE OF X IS
 1.0000475E+00        1.0000475E+00        1.0000475E+00        1.0000475E+00
THE CONSTRAINT VALUES
                      NOT INCLUDING THE NON-NEGATIVITIES
 1.9083354E-04        9.6102526E-05
APPARENTLY ROUNDOFF ERRORS PREVENT A MORE ACCURATE DETERMINATION OF THE MINIMUM OF THIS SUBPROBLEM.

2ND  ORDER ESTIMATES
```

```
  F=  3.9999870E+00      P=  4.0004638E+00      G=  3.9999872E+00      RSIGMA=  0.                    H=  0.
 THE CURRENT VALUE OF X IS
  9.9999677E-01        9.9999676E-01        9.9999676E-01        9.9999676E-01
 THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
 -1.2096163E-05       -5.3559433E-06


1ST  ORDER ESTIMATES
  F=  3.9999872E+00      P=  4.0005105E+00      G=  3.9999872E+00      RSIGMA=  0.                    H=  0.
 THE CURRENT VALUE OF X IS
  9.9999681E-01        9.9999680E-01        9.9999680E-01        9.9999680E-01
 THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
 -1.1930379E-05       -5.2730559E-06


LAGRANGE MULTIPLIERS
  F=  3.9999872E+00      P=  4.0005105E+00      G=  3.9999872E+00      RSIGMA=  0.                    H=  0.
 THE CURRENT VALUE OF X IS
  9.9995250E-05        9.9995252E-05        9.9995252E-05        9.9995252E-05
 THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
  5.2401690E-01        1.0405554E+00
 TIME=  315.442 SECONDS
*****************************************
      POINT=  18        DOTT=  3.2081309E+00      RHO=  1.0000000E-05      MAGNITUDE=  1.7911256E+00      PHASE= 2
  F=  4.0000197E+00      P=  4.0002418E+00      G=  3.9999597E+00      RSIGMA=  2.2209197E-04      H=  0.
 THE CURRENT VALUE OF X IS
  1.0000049E+00        1.0000049E+00        1.0000049E+00        1.0000049E+00
 THE CONSTRAINT VALUES
                        NOT INCLUDING THE NON-NEGATIVITIES
  2.0593770E-05        1.0988132E-05
```

REFERENCES

1. Bracken, J. and McCormick, G.P., Selected Applications of Nonlinear Programming, Wiley, 1968.

2. Bracken, J. and McGill, J.T., "Mathematical Programs With Optimization Problems in the Constraints", Institute for Defense Analyses, Research Paper P-725, July 1971.

3. Bracken, J. and McGill, J.T., "Mathematical Programs With Optimization Problems in the Constraints: Applications to Defense Analyses", Institute for Defense Analyses, Research Paper P-784, August 1971 (DRAFT).

4. Fiacco, A.V. and McCormick, G.P., Nonlinear Programming: Sequential Unconstrained Minimization Techniques, Wiley, 1968.

5. McCormick, G.P., Mylander, W.C. and Fiacco, A.V., "Computer Program Implementing the Sequential Unconstrained Minimization Technique for Nonlinear Programming", Research Analysis Corporation, Technical Paper TP-151, April 1965.

6. Mylander, W.C., Holmes, R.L. and McCormick, G.P., "A Guide to SUMT - Version 4: The Computer Program Implementing the Sequential Unconstrained Minimization Technique for Nonlinear Programming", Research Analysis Corporation, RAC-P-63, October 1971.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*
PROGRAM ANALYSIS DIVISION
INSTITUTE FOR DEFENSE ANALYSES

2a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

2b. GROUP

3. REPORT TITLE
COMPUTER PROGRAM FOR SOLVING MATHEMATICAL PROGRAMS WITH NONLINEAR PROGRAMS IN THE CONSTRAINTS

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

5. AUTHOR(S) *(First name, middle initial, last name)*
JEROME BRACKEN AND JAMES T. McGILL

6. REPORT DATE
March 1972

7a. TOTAL NO. OF PAGES
107

7b. NO. OF REFS
6

6a. CONTRACT OR GRANT NO.

b. PROJECT NO.

c. IDA Independent Research Program

d.

9a. ORIGINATOR'S REPORT NUMBER(S)
P-801

9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

10. DISTRIBUTION STATEMENT
This document is unclassified and suitable for public release.

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY
N/A

13. ABSTRACT

This paper documents a computer program to be used in solving nonlinear programming problems with nonlinear programming problems in the constraints. The program, named INSUMT, is used with the standard program, named SUMT, which implements the sequential unconstrained minimization technique for nonlinear programming. SUMT calls INSUMT when it is necessary to solve a nonlinear program in a constraint. The INSUMT program, together with a fairly complete example of its use, is included in the documentation.

Theory and applications of the models which can be solved using this program are treated in two companion papers.

DD FORM 1 NOV 65 1473

UNCLASSIFIED

Security Classification

UNCLASSIFIED

Security Classification

14 KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
OPTIMIZATION						
NONLINEAR PROGRAMMING						
MAX-MIN						
TWO-SIDED OPTIMIZATION						
GAME THEORY						
SUMT						
INSUMT						
COMPUTER PROGRAM						
ALGORITHM						

UNCLASSIFIED

Security Classification

U145548