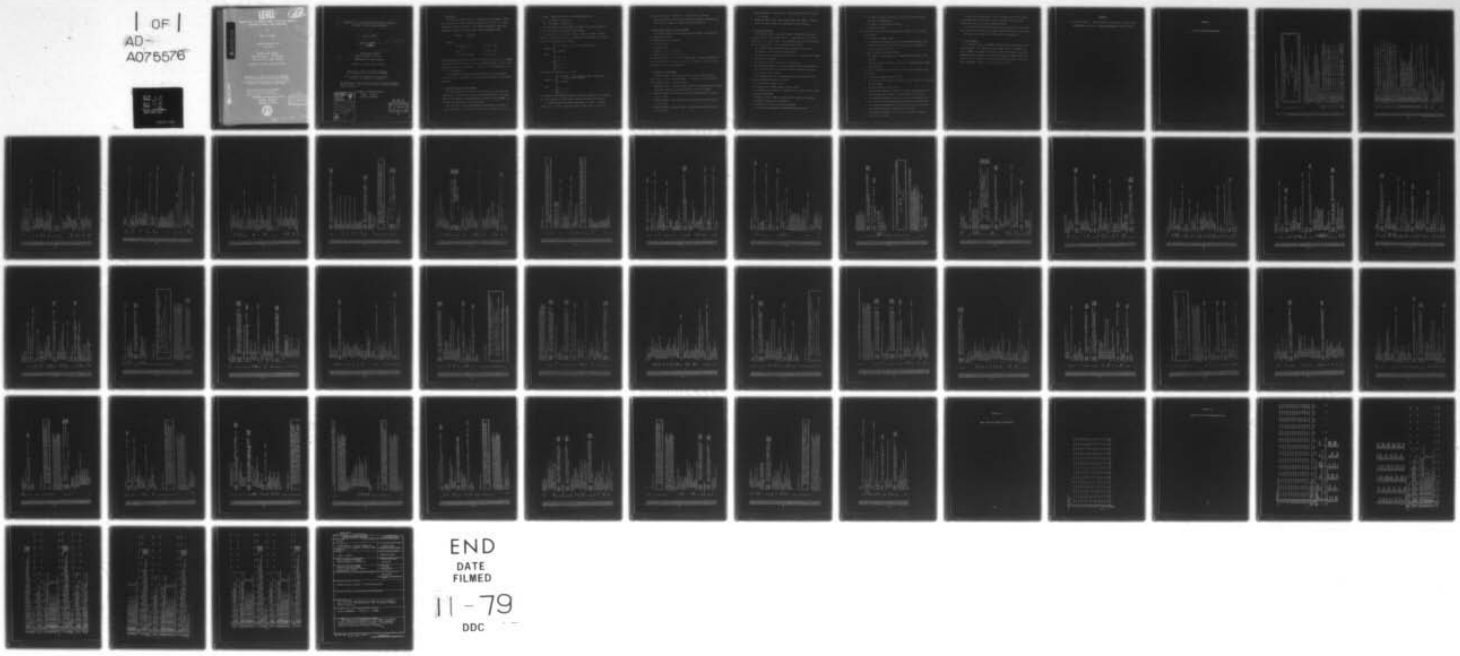
AD-A075 576 STANFORD UNIV CA DEPT OF OPERATIONS RESEARCH F/G 9/2
DOCUMENTATION OF A COMPUTER PROGRAM FOR HILLIER'S HEURISTIC PRO--ETC(U)
JUL 79 N E JACQMI N00014-76-C-0418
UNCLASSIFIED TR-88 NL

| OF |
AD-
A075576



END
DATE
FILMED
11-79
DDC

LEVEL II

12

DOCUMENTATION OF A COMPUTER PROGRAM FOR HILLIER'S HEURISTIC PROCEDURE IN INTEGER LINEAR PROGRAMMING

AD A075576

BY

NANCY E. JACQMIN

TECHNICAL REPORT NO. 88
JULY 1979

PREPARED UNDER CONTRACT
N00014-76-C-0418 (NR-047-061)
FOR THE OFFICE OF NAVAL RESEARCH

Frederick S. Hillier, Project Director

Reproduction in Whole or in Part is Permitted
for any Purpose of the United States Government

This document has been approved for public release
and sale; its distribution is unlimited.

DDC FILE COPY

THIS DOCUMENT IS BEST QUALITY PRACTICABLE.
THE COPY FURNISHED TO DDC CONTAINED A
SIGNIFICANT NUMBER OF PAGES WHICH DO NOT
REPRODUCE LEGIBLY.

DEPARTMENT OF OPERATIONS RESEARCH
STANFORD UNIVERSITY
STANFORD, CALIFORNIA



DDC
RECEIVED
OCT 29 1979
D

79 10 26 019

DOCUMENTATION OF A COMPUTER PROGRAM FOR HILLIER'S HEURISTIC PROCEDURE IN INTEGER LINEAR PROGRAMMING

by

NANCY E. JACQMIN

TECHNICAL REPORT NO. 88

JUL[illegible]79

TR-88

58

PREPARED UNDER CONTRACT

N00014-76-C-0418 (NR-047-061)

NSF-MCS76-81259

FOR THE OFFICE OF NAVAL RESEARCH

Frederick S. Hillier, Project Director

Reproduction in Whole or in Part is Permitted
for any Purpose of the United States Government

This document has been approved for public release
and sale; its distribution is unlimited.

This research was supported in part by National Science Foundation
Grant NSF Number - MCS. 76-81259 A01, Dept. of Operations Research
Stanford University.

DEPARTMENT OF OPERATIONS RESEARCH
STANFORD UNIVERSITY
STANFORD, CALIFORNIA

Accession For
NTIS GRA&I
DDC TAB
Unannounced
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or special
A

DDC
RECEIVED
OCT 29 1979
D

402 766

1. Introduction

The computer code HEUR, written in FORTRAN by Bruce H. Faaland to implement the heuristic procedure developed by Frederick S. Hillier [1], is documented and listed in this report. This program is designed to find a good approximate solution to the pure integer linear programming problem,

$$\text{maximize} \quad x_0 = \sum_{j=1}^{n} c_j x_j,$$

subject to

$$(1) \quad \sum_{j=1}^{n} a_{ij} x_j \leq b_i \qquad (i = 1,2,\ldots,m),$$

$$(2) \quad x_j \geq 0 \qquad (j = 1,2,\ldots,n),$$

$$(3) \quad x_j \text{ is an integer} \qquad (j = 1,2,\ldots,n).$$

This problem may also be written as max cx subject to $Ax \leq b$, $x \geq 0$, x integer where the constraint matrix A is $m \times n$, the right hand side b is an $m \times 1$ column vector, the cost coefficients c form a $1 \times n$ row vector and x is an $n \times 1$ column vector.

It is assumed that the set of points satisfying (1) and (2) possesses an interior point. See Hillier [1] for a detailed discussion of this heuristic procedure.

2. Main Program and Input Requirements

The main program for the heuristic code HEUR serves to coordinate the three phases of the search for the desired approximate solution. Any combination of the methods for these phases may be accessed by the programmer via the IFLAG and LFLAG arrays to be described later.

The first card of input contains NAME, any alphanumeric characters to identify the problem. The second card contains the variables M and N in

2I5 format. These variables have the following significance:

M = the number of rows in A.

N = the number of columns in A.

The third input card contains RPT, the number of extra (not necessarily distinct) solutions to be obtained, in F5.0 format.

The fourth input card contains IFLAG(1), IFLAG(2), IFLAG(3), IFLAG(4), LFLAG(1,1), LFLAG(1,2), LFLAG(2,1), and LFLAG(2,2) in 8I4 format. The significance of these variables is as follows:

$$\text{IFLAG}(I) = \begin{cases} 1, \text{ if criterion } I' \text{ is to be used in Phase 2} \\ 0, \text{ otherwise} \end{cases}$$

where

$$I' = \begin{cases} B, \text{ if } I = 1 \\ C, \text{ if } I = 2 \\ A, \text{ if } I = 3 \\ D, \text{ if } I = 4. \end{cases}$$

A,B,C and D are as defined in [1, p. 608].

$$\text{LFLAG}(I,J) = \begin{cases} 1, \text{ if method } I \text{ is to be used in Phase 1 and method } J' \text{ is to be used in Phase 2} \\ 0, \text{ otherwise} \end{cases}$$

where

$$J' = \begin{cases} 3, \text{ if } J = 1 \\ 2, \text{ if } J = 2. \end{cases}$$

A description of the methods for Phases 1 and 2 may be found in [1, pp. 605-607]. Note that any number of combinations can be run as specified on this card.

The remaining input cards contain the arrays A, b and c in 15F5.0 format. The A matrix is read in one row at a time, then b is read in

starting on a new card, followed by c also starting on a new card.

Another problem may then be input using the card sequence described above.

The input for a sample run of HEUR is given in Appendix II.

3. Restrictions Relevant to the Use of HEUR

As currently written, the following restrictions apply to the input data and problem size for HEUR:

1. NAME must have $\leq$ 20 characters.
2. M must be $\leq$ 120.
3. N must be $\leq$ 120.
4. IFLAG(I) = 0 or 1, I = 1,2,3,4.
5. LFLAG(I,J) = 0 or 1, I = 1,2; J = 1,2.

To change the bounds on M and N you simply have to modify the dimension of A from A(121,362) to A(m' + 1, n' + 2m' + 2) where m' and n' are the new bounds on M and N respectively.

4. Description of Subprograms

Detailed comments on the specific function of each subroutine may be found in the listing of HEUR in Appendix I. Variables of note are defined in these comments as well. In general terms:

1. SUBROUTINE SIMPLX -- Finds an optimal solution to the stated problem ignoring the integer constraints (this is referred to as the "LP problem").
2. SUBROUTINE PHTW01 -- Executes Phase 2 using variable selection criterion B. [1, p. 608]
3. SUBROUTINE PHTW02 -- Executes Phase 2 using variable selection criterion C. [1, p. 608]
4. SUBROUTINE PHTW03 -- Executes Phase 2 using variable selection criterion A. [1, p. 608]

5. SUBROUTINE PHTWO4 -- Executes Phase 2 using variable selection criterion D. [1, p. 608]

6. SUBROUTINES PART1, PART2, PART3, PART4, PART5, PART6, PART7 -- Together execute the parts of Phase 3 as described in [1, pp. 612-616].

5. Description of Output

The following output is generated by HEUR. The NAME of the problem is printed followed by the constraint matrix A, right hand side b and cost coefficients c in 15F8.2 format. The normalization factor for the objective function $(1/(\sum_{j=1}^{n} c_j^2)^{1/2})$ is also recorded in F10.2 format.

On return from SUBROUTINE SIMPLEX the following are printed out:

1. The solution to the LP problem in 15F8.2 format.
2. The row that each original variable is basic in for this solution (0 means the variable is nonbasic).
3. The indices of the basic variables for this solution.
4. The normalized optimal basis inverse (the basis inverse corresponding to the LP solution when the constraints have been normalized as defined in [1, p, 605]) in 6F13.5 format.

Then for each combination of methods specified by the LFLAG and IFLAG arrays the following information is printed:

1. The methods used in Phases 1 and 2.
2. The criterion for variable selection used in Phase 2.
3. The starting value of ALPHA (α as defined in [1, p. 607]) for the current solution being sought.
4. The Phase 1 solution XTWO (the point which with XONE, the LP solution, determines the search line for Phase 2).
5. The results of the Phase 2 search which include:
   a) The value of ALPHA where the feasible solution was found.

b) The number of points on the search line that were moved to during the search (excluding XONE).

c) The number of trial solutions examined.

d) The Phase 2 solution.

6. The results of the first time through Mode 1 of Phase 3 [1, pp. 611-613] which include:

a) The number of solution changes.

b) The normalized improvement (the actual improvement multiplied by the normalization factor for the objective function) in the objective function value in F8.4 format.

7. A record of Phase 3 execution including:

a) The number of times through each of SUBROUTINES PART4, PART5, PART6 and PART7.

b) The number of solution changes during each of SUBROUTINES PART4, PART5, PART6 and PART7.

8. At the end of execution of the heuristic procedure the following results are printed:

a) The final approximate integer solution.

b) The feasibility test slacks (as defined in Part I, Step 3 of [1, p. 612]) in 15F8.2 format.

c) The final objective function value.

d) The normalized difference in objective function value between the LP solution (XONE) and the Phase 1 solution (XTWO) in F10.4 format.

e) The normalized difference in objective function value between the LP solution and the final approximate integer solution in F10.4 format.

f) The normalized improvement in the objective function value since the first time through Mode 1 of Phase 3 in F10.4 format.

g) The normalized improvement in the objective function value during Phase 3 in F10.4 format.

If RPT > 0.0 the information in items 1 through 8 above will be printed out for each extra solution found, as well as the initial one, for every combination of Phase 1 and Phase 2 methods specified.

If more than one problem was included in the input data all of the information above will be printed out for each problem starting with its input NAME.

The output from a sample run of HEUR is given in Appendix III.

6. Further Comments

As currently written, the FORTRAN code HEUR contains approximately 2,300 source statements. All significant array storage is done in a common block labeled COMMON. When compiled on Stanford's IBM 370/168 computer under IBM's FORTRAN-H-Extended compiler, with level two optimization, approximately 512K bytes of core are required for all instruction and data storage. The code is WATFIV compatible. HEUR was last revised in October 1978.

REFERENCE

1. Hillier, Frederick S., "Efficient Heuristic Procedures for Integer Linear Programming with an Interior," Operations Research 17, 600-637 (1969).

APPENDIX I

LISTING OF COMPUTER PROGRAM HEUR

```
//NEJME JR     JOB  NEJ$##,'JACOMIN, N',REGION=512K,TIME=(1,0)
//  EXEC   FORTHCG,PARM.FORT='OPT=2'
//FORT.SYSIN DD *
C
C
C     ****************************************************************************
C     ****************************************************************************
C     **     THE PURPOSE OF THIS CODE IS TO FIND A GOOD APPROXIMATE SOLUTION    **
C     **     OF THE PURE INTEGER LINEAR PROGRAMMING PROBLEM:                    **
C     **                                                                        **
C     **                          MAXIMIZE    XZERO = CX                        **
C     **                  SUBJECT TO:                                           **
C     **                              AX . LE . B                               **
C     **                               X . GE . 0                               **
C     **                          ALL COMPONENTS OF X INTEGRAL                  **
C     **                                                                        **
C     **     WHERE: THE DIMENSION OF A IS M BY N                                **
C     **            THE DIMENSION OF B IS M                                     **
C     **            THE DIMENSION OF C IS N                                     **
C     **     AND THE SET OF LP SOLUTIONS IS ASSUMED TO HAVE AN INTERIOR POINT.  **
C     ****************************************************************************
C     ****************************************************************************
C
C
      DIMENSION A(121,362),B(121),C(121),CAPR(121,121),CPERM(121),
     XDELX(121),DELTA(121),JPERM(121),LQ(121),LKB(362),
     XQ(121),QSTAR(121),S(121),SPRIME(121),X(121),
     XXL(121),XONE(121),XTWO(121),NEGB(121)
      DIMENSION BB(121,121),BTWO(121),INFIX(8),TOL(4),KOUT(7),ERR(8),
     XJH(121),P(121),Y(121),KB(362)
      DIMENSION NAME(20)
      DIMENSION IFLAG(8),LFLAG(2,2)
      DIMENSION XSAVE(121)
      COMMON A,ALPHA,B,C,CAPR,CPERM,DELX,I,IMPROV,DELTA,
     XINVEST,J,JHIGH,JJ,JPERM,K,KEY,LESS,LJ,LK,LL,LM,LML,LQ,LTEMP,M,
     XMETHOD,NLESS,NZERO,PHI,Q,QMAX,QSTAR,S,SLPRIM,SLSUBK,
     XSPRIME,SUM,T,TEMP,TT,X,XL,XONE,XTWO,USUBK,ZM,ZZ,N,
     XLSOLN2,LSOLN4,LSOLN5,LSOLN6,LSOLN7
200   FORMAT(16F5.0)
201   FORMAT(2I5)
202   FORMAT('0',' THE NORMALIZED INVERSE OF THE OPTIMAL BASIS IS:')
447   FORMAT('0',' THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNC',
     X'TION VALUE SINCE THE FIRST TIME THROUGH MODE 1 = ',T95,F10.4,'.')
450   FORMAT('0',' THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE ',
     X'FUNCTION VALUE IN PHASE 3 = ',T95,F10.4,'.')
451   FORMAT('0',' THE NORMALIZED DIFFERENCE BETWEEN THE OPTIMAL ',
     X'LP AND INTEGER OBJECTIVE FUNCTION VALUES = ',T95,F10.4,'.')
454   FORMAT(' ',' THE NUMBER OF SOLUTION CHANGES DURING PART 4 = ',
     XT55,I8,'.')
455   FORMAT(' ',' THE NUMBER OF SOLUTION CHANGES DURING PART 5 = ',
     XT55,I8,'.')
456   FORMAT(' ',' THE NUMBER OF SOLUTION CHANGES DURING PART 6 = ',
     XT55,I8,'.')
457   FORMAT(' ',' THE NUMBER OF SOLUTION CHANGES DURING PART 7 = ',
     XT55,I8,'.')
464   FORMAT('0',' THE NUMBER OF TIMES THROUGH PART 4 = ',T55,I8,'.')
465   FORMAT('0',' THE NUMBER OF TIMES THROUGH PART 5 = ',T55,I8,'.')
466   FORMAT('0',' THE NUMBER OF TIMES THROUGH PART 6 = ',T55,I8,'.')
467   FORMAT('0',' THE NUMBER OF TIMES THROUGH PART 7 = ',T55,I8,'.')
```

```
  472  FORMAT('0',*FOR THE FIRST TIME THROUGH MODE 1 WE HAVE THE FOL',
 X*LOWING DATA:')
  473  FORMAT(' ',' THE NUMBER OF SOLUTION CHANGES = ',T65,I8,'.')
  475  FORMAT(' ',' THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE ',
 X*FUNCTION VALUE = ',T65,F8.4,'.')
  469  FORMAT('-','THE FOLLOWING OCCURRED PRIOR TO THE TERMINATION OF ',
 X*PHASE 3:')
  476  FORMAT('0',' DURING PHASE 3 WE ENTERED MODE 1',I4,' TIMES AND HAD'
 X,I4,' SOLUTION CHANGES.')
  483  FORMAT('-',*AT THE END OF THE HEURISTIC PROCEDURE WE HAVE THE ',
 X*FOLLOWING RESULTS:')
  484  FORMAT('0',' THE NORMALIZED DIFFERENCE BETWEEN THE XONE AND XTWO',
 X* OBJECTIVE FUNCTION VALUES = ',T95,F10.4,'.')
  477  FORMAT('-','AT THE END OF PHASE 2 WE HAVE THE FOLLOWING DATA:')
  485  FORMAT('0',' THE VALUE OF ALPHA WHERE THE FEASIBLE SOLUTION ',
 X*WAS FOUND) = ',T65,F8.4,'.')
  486  FORMAT('0',' THE NUMBER OF POINTS MOVED TO ON THE SEARCH LINE',
 X* = ',T65,I8,'.')
  487  FORMAT('0',' THE NUMBER OF TRIAL SOLUTIONS BEFORE FINDING THE '
 X,'FEASIBLE SOLUTION = ',T65,I8,'.')
  500  FORMAT(1X,15F8.2)
 6001  FORMAT(1X,6F13.5)
  501  FORMAT('0',' THE FINAL APPROXIMATE INTEGER SOLUTION IS:')
  502  FORMAT('0',' THE SOLUTION FROM PHASE 2 IS:')
  503  FORMAT('0','THE CONSTRAINT MATRIX A(I,J) IS:')
  504  FORMAT('0','THE RIGHT HAND SIDE B(I) IS:')
  505  FORMAT('0','THE COST COEFFICIENTS C(J) ARE:')
  438  FORMAT('-','THE SIMPLEX METHOD GIVES US:')
  506  FORMAT('0',' THE LP SOLUTION XONE IS:')
  507  FORMAT('-','THE PHASE 1 SOLUTION XTWO IS:')
  508  FORMAT('0',' THE FEASIBILITY TEST SLACKS ARE:')
  509  FORMAT(' ','THE CURRENT VALUE OF ALPHA IS',F10.2,'.')
  510  FORMAT('0','THE NORMALIZATION FACTOR FOR THE OBJECTIVE FUNCTION ',
 X*IS',F10.2,'.')
  511  FORMAT(' ',*PHASE 2, CRITERION',F3.0)
  571  FORMAT(1X,15I8)
  573  FORMAT('1',*PHASE 1, METHOD',F3.0,'  PHASE 2, METHOD',F3.0)
  574  FORMAT(20A4)
  575  FORMAT(1H1,5X,20A4)
  576  FORMAT('0',' FOLLOWING, IN ORDER OF INCREASING J, IS THE RO* ',
 X*X(J) IS BASIC IN:')
  577  FORMAT('0',' FOLLOWING, IN ORDER OF INCREASING I, IS THE ITH ',
 X*BASIC VARIABLE, INCLUDING SLACKS:')
 5001  FORMAT(6F13.5)
 5711  FORMAT(15I4)
 5712  FORMAT('0',' THE OBJECTIVE FUNCTION VALUE OF THE FINAL SOLUTION',
 X' = ',T95,F10.2,'.')
C
C     *****     READ IN DATA.       *****
C
 1000 READ(5,574,END=2300) (NAME(I),I=1,20)
      WRITE(6,575) (NAME(I),I=1,2))
      READ(5,201) M,N
      READ(5,200) RPT
      READ(5,5711)(IFLAG(I),I=1,4),((LFLAG(I,J),J=1,2),I=1,2)
      COUNT=0.0
      DO 183 I=1,M
  183   READ(5,200)(A(I,J),J=1,N)
      READ(5,200)(B(I),I=1,M)
      READ(5,200)(C(I),I=1,N)
```

```
C
      WRITE(6,503)
      DO 132 I=1,M
  132   WRITE(6,500)(A(I,J),J=1,N)
      WRITE(6,504)
      WRITE(6,500)(B(I),I=1,M)
      WRITE(6,505)
      WRITE(6,500)(C(I),I=1,N)
C
C     *****       NORMALIZE COEFFICIENTS OF PROBLEM.      *****
C
      DO 400 I=1,M
        SUM=0.0
        DO 401 J=1,N
  401     SUM=SUM+A(I,J)*A(I,J)
        SUM=SQRT(SUM)
        DO 402 J=1,N
          A(I,J)=A(I,J)/SUM
  402     CONTINUE
        B(I)=B(I)/SUM
  400   CONTINUE
      CNORM=0.0
      DO 410 J=1,N
        CNORM=CNORM+C(J)*C(J)
  410   CONTINUE
      CNORM=1.0/SQRT(CNORM)
      WRITE(6,510)CNORM
C
C     *****       THE FOLLOWING SETS UP THE DATA FOR LP ROUTINE.     *****
C
      LK=N+1
      LL=M+1
      LM=M+N
C
C     *****       STORE COST COEFFICIENTS IN FIRST ROW OF A.     *****
C
      DO 157 I=1,M
        I1=M-I+1
        I2=M-I+2
        LVL=I+1
        QSTAR(LVL)=B(I)
        DO 156 J=1,N
          A(I2,J)=A(I1,J)
  154     CONTINUE
  157   CONTINUE
      QSTAR(1)=0.0
      DO 150 J=1,N
        A(1,J)=-C(J)
  150   CONTINUE
C
C     *****       ADD SLACK VARIABLES.          *****
C
      DO 151 J=LK,LM
        A(1,J)=0.0
  151   CONTINUE
      DO 152 I=2,LL
        DO 153 J=LK,LM
          A(I,J)=0.0
  153     CONTINUE
  152   CONTINUE
      DO 154 I=2,LL
```

```
180.                J=N+I-1
181.                A(I,J)=1.0
182.      154       CONTINUE
183.      C
184.      C       *****       ADD ARTIFICIAL VARIABLES WHEN B(I) . LT . 0.     *****
185.      C
186.              FINALN=N+M
187.              DO 9995 I=1,M
188.      9995      NEGB(I)=0.0
189.              T=LM
190.              DO 9990 I=2,LL
191.                IF(OSTAR(I).GE.0.)GO TO 9990
192.                T=T+1
193.                DO 9991 J=1,LM
194.      9991        A(I,J)=-A(I,J)
195.                QQ=I-1
196.                NEGB(QQ)=1.0
197.                B(QQ)=-B(QQ)
198.                OSTAR(I)=-OSTAR(I)
199.                DO 9992 J=2,LL
200.      9992        A(J,T)=0.0
201.      C
202.      C       *****   USE BIG M METHOD ON ARTIFICIAL VARIABLES.     *****
203.      C
204.                A(1,T)=1.0
205.                A(1,T)=1000
206.      9990      CONTINUE
207.      C
208.      C       *****  FINALN CONTAINS NUMBER OF VARIABLES FOR LP ROUTINE.  *****
209.      C
210.              FINALN=T
211.              INFIX(1)=4
212.              INFIX(2)=FINALN
213.              INFIX(3)=121
214.              INFIX(4)=441
215.              INFIX(5)=2
216.              INFIX(6)=1
217.              INFIX(7)=500
218.              INFIX(8)=0
219.              TOL(1)=10.0**(-5)
220.              TOL(2)=10.0**(-5)
221.              TOL(3)=-10.0**(-3)
222.              TOL(4)=10.0**(-13)
223.              PRM=0.0
224.      C
225.      C       *****      FIND LP SOLUTION.      *****
226.      C
227.              CALL SIMPLX(INFIX,A,OSTAR,TOL,PRM,KOUT,ERR,JH,X,P,Y,KR,BB)
228.      C
229.      C       *****  THE FOLLOWING RESTORES THE A MATRIX TO ORIGNAL FORM. *****
230.      C
231.              DO 1103 I=1,M
232.                LML=I+1
233.                DO 1104 J=1,N
234.                  A(I,J)=A(LML,J)
235.      1104      CONTINUE
236.      1103    CONTINUE
237.      C
238.      C       *****  THE FOLLOWING RESTORES ROWS OF INVERSE AND A    *****
239.      C       *****  NEGATED DUE TO NEGATIVE B(I).                   *****
```

```
      C
            DO 9993 I=1,M
              IF(NEGB(I).LE.0.)GO TO 9993
              DO 9994 J=1,N
 9994           A(I,J)=-A(I,J)
              DO 9996 K=2,LL
                II=I+1
 9996           BB(II,K)=-BB(II,K)
              B(I)=-B(I)
 9993         CONTINUE
            DO 900 I=1,LL
  900         XSAVE(I)=X(I)
      C
      C     *****  STORE LP SOLUTION IN XONE.      *****
      C
            DO 155 J=1,N
              LL=KB(J)
              IF(LL.GT.0) GO TO 156
              XONE(J)=0.0
              GO TO 155
  156         XONE(J)=X(LL)
  155         CONTINUE
            WRITE(6,498)
            WRITE(6,506)
            WRITE(6,6001)(XONE(I),I=1,N)
            LL=M+1
            DO 578 J=1,N
              IF(KB(J).LE.0)GO TO 579
              LKB(J)=KB(J)-1
              GO TO 578
  579         LKB(J)=0
  578         CONTINUE
            WRITE(6,576)
            WRITE(6,571)(LKB(J),J=1,N)
            WRITE(6,577)
            WRITE(6,571)(JH(I),I=2,LL)
            JJ=1
            II=0
      C
      C     *****   RETRIEVE NORMALIZED OPTIMAL BASIS INVERSE.     *****
      C
            DO 160 J=1,LL
              DO 161 I=1,LL
                II=II+1
                IF(II.LE.LL)GO TO 162
                II=1
                JJ=JJ+1
  162           CONTINUE
                A(I,N+J)=BB(II,JJ)
  161           CONTINUE
  160         CONTINUE
            DO 163 I=1,M
              DO 164 J=1,M
                BB(I,J)=A(I+1,N+J+1)
  164           CONTINUE
  163         CONTINUE
            WRITE(6,202)
            WRITE(6,5001)((BB(I,J),J=1,M),I=1,M)
 1232       CONTINUE
      C
```

```
C     *****  IFLAG(I)=1 MEANS TO USE CRITERION I IN PHASE 2.      *****
C     *****  LFLAG(I,J)=1 MEANS TO USE METHOD I IN PHASE 1 AND    *****
C     *****  METHOD J IN PHASE 2.                                *****
C
      COUNT=COUNT+1.0
      ICOUNT=COUNT
      IF(IFLAG(ICOUNT).EQ.0) GO TO 1932
      KPH=1
      KPH1=1
      IF(LFLAG(KPH,KPH1).EQ.0) GO TO 1899
      GO TO 169
165   KPH=1
      KPH1=2
      IF(LFLAG(KPH,KPH1).EQ.0) GO TO 1899
      GO TO 169
166   KPH=2
      KPH1=1
      IF(LFLAG(KPH,KPH1).EQ.0) GO TO 1899
      GO TO 169
167   KPH=2
      KPH1=2
      IF(LFLAG(KPH,KPH1).EQ.0)GO TO 1899
169   CONTINUE
      PH=KPH
      PH1=KPH1
      ALPHA=0.0
      REPEAT=0.0
1969  CONTINUE
C
C     *****  PRINT OUT OPTIONS TO BE USED AND PRESENT      *****
C     *****  VALUE OF ALPHA.  FOR EACH INPUT PROBLEM THIS *****
C     *****  SECTION WILL BE ENTERRED (RPT+1) TIMES.       *****
C
      WRITE(6,573)PH,PH1
      WRITE(6,511)COUNT
      WRITE(6,509)ALPHA
      LL=M+1
      DO 901 I=1,LL
901     X(I)=XSAVE(I)
C
C
C     ***************************************************************
C     *     THIS IS PHASE 1.  FIND A POINT NEAR XONE BUT WELL WITHIN THE      *
C     *     INTERIOR OF THE FEASIBLE LP REGION WHICH CAN BE ROUNDED TO THE    *
C     *     NEAREST INTEGER SOLUTION WITHOUT VIOLATING THE CONSTRAINTS THAT   *
C     *     ARE BINDING AT XONE.                                              *
C     ***************************************************************
C
      IF(PH.EQ.2.0) GO TO 170
C
C     *****     THIS IS METHOD 1 OF PHASE 1.  REPLACE B(I) BY      *****
C     *****                                                        *****
C     *****     BTWO(I) = B(I) - 1/2(SUM OVER J IN 1,...,N AND     *****
C     *****                          BASIC AT XONE ABS(A(I,J)))    *****
C     *****                FOR ALL I BINDING AT XONE.              *****
C
      DO 172 I=1,M
        LL=KB(N+I)
        IF(LL.LE.0)GO TO 11
```

```
            IF(X(LL).GT.0.0)GO TO 174
   11       TEMP=0.0
            DO 173 J=1,N
              IF(KB(J).GT.0) TEMP=TEMP+ABS(A(I,J))
  173         CONTINUE
            BTWO(I)=B(I)-(.5)*TEMP
            GO TO 172
  174       BTWO(I)=B(I)
  172       CONTINUE
          GO TO 175
  170     CONTINUE
C
C         *****     THIS IS METHOD 2 OF PHASE 1.  REPLACE B(I) BY      *****
C         *****                                                        *****
C         *****     BTWO(I) = B(I) - 1/2(SQRT(NUMBER OF VARIABLES      *****
C         *****                              THAT ARE BASIC AT XONE    *****
C         *****                              EXCLUDING SLACKS AND ARTI-*****
C         *****                              FICIALS))                 *****
C         *****                    FOR ALL I BINDING AT XONE.          *****
C
          NN=0
          DO 176 J=1,N
            IF(KB(J).GT.0) NN=NN+1
  176       CONTINUE
          T=NN
          TEMP=(.5)*(SQRT(T))
          DO 177 I=1,M
            LL=KB(N+I)
            IF(LL.LE.0)GO TO 12
            IF(X(LL).GT.0.0)GO TO 178
   12       BTWO(I)=B(I)-TEMP
            GO TO 177
  178       BTWO(I)=B(I)
  177       CONTINUE
  175     CONTINUE
C
C         *****   SOLVE A*XTWO = BTWO FOR THE SAME BASIS AS XONE.    *****
C
          DO 179 J=1,N
            XTWO(J)=0.0
  179       CONTINUE
          DO 180 I=1,M
            LL=JH(I+1)
            IF(LL.LE.0)GO TO 180
            IF(LL.GT.N)GO TO 180
            TEMP=0.0
            DO 181 K=1,M
              TEMP=TEMP+BB(I,K)*BTWO(K)
  181         CONTINUE
            XTWO(LL)=TEMP
  180       CONTINUE
          WRITE(6,507)
          WRITE(6,500)(XTWO(J),J=1,N)
C
C         *****   THE FOLLOWING CHECKS WHICH VERSION OF PHASE 2  *****
C         *****   IS TO BE USED.                                 *****
C
          METHOD=1
          LJ=0
          LK=0
```

```
      ZM=REPEAT
      IF(COUNT.GT.4.0) RETURN
C
C
C     ****************************************************************
C     *      THIS IS ENTRY TO PHASE 2.  MOVE SLOWLY DOWN THE LINE SEGMENT  *
C     *      JOINING XONE AND XTWO SEARCHING FOR A NEARBY FEASIBLE INTEGER  *
C     *      SOLUTION.                                                 *
C     ****************************************************************
C
C
      IF(COUNT.EQ.1.0)CALL PHTWO1
      IF(COUNT.EQ.2.0) CALL PHTWO2
      IF(COUNT.EQ.3.0) CALL PHTWO3
      IF(COUNT.EQ.4.0) CALL PHTWO4
C
C     *****    PRINT OUT RESULTS OF PHASE 2 SEARCH.    *****
C
      WRITE(6,477)
      WRITE(6,485)ALPHA
      WRITE(6,486)LK
      WRITE(6,487)LJ
      WRITE(6,502)
      WRITE(6,500)(X(LML),LML=1,N)
C
C     *****    STORE PHASE 2 OBJECTIVE VALUE IN ZPH2.    *****
C
      ZPH2=0.0
      DO 906 LML=1,N
  906   ZPH2=ZPH2+X(LML)*C(LML)
      IF(METHOD).EQ.0) GO TO 1899
C
C
C     ****************************************************************
C     *      THIS IS PHASE 3.  GIVEN THE FEASIBLE SOLUTION FROM PHASE 2, SEARCH *
C     *      FOR A BETTER ONE.  THIS SEARCH IS CONDUCTED BY TWO ALTERNATING  *
C     *      MODES.                                                    *
C     ****************************************************************
C
C
      CALL PART1
      IF(NZERO.LE.1)GO TO 903
      CALL PART3
  903 CONTINUE
      NBR2=0
      NBR5=0
      NBR4=0
      NBR6=0
      NBR7=0
      LSOLN2=0
      LSOLN4=0
      LSOLN5=0
      LSOLN6=0
      LSOLN7=0
   27 CONTINUE
      CALL PART2
      NBR2=NBR2+1
      IF(NBR2.GT.1)GO TO 27
C
C
```

```
C     *****   PRINT OUT RESULTS OF FIRST TIME THROUGH   *****
C     *****   PHASE 1, PART 2 (MODE 1).                 *****
C
C
      WRITE(6,472)
      WRITE(6,473)LSOLN2
C
C     *****   STORE INITIAL MODE 1 OBJECTIVE VALUE IN ZPRT2.   *****
C
      ZPRT2=0.0
      DO 907 LML=1,N
  907   ZPRT2=ZPRT2+C(LML)*X(LML)
      TEMP=CNORM*(ZPRT2-ZPR42)
      WRITE(6,475)TEMP
      IF(NZERO.LE.1)GO TO 904
   27 CONTINUE
C
C     *****   STORE CURRENT MODE 1 SOLUTION IN XL.   *****
C
      DO 21 J=1,N
        XL(J)=X(J)
   21   CONTINUE
      J=1
      K=N
   30 CONTINUE
      CALL PART4
      NBP4=NBP4+1
      IF(INVEST.EQ.1)GO TO 40
      K=J+1
      GO TO 80
C
C     *****   LOOK TO SEE IF A SIMULTANEOUS CHANGE IN X(JPERM(J))   *****
C     *****   AND X(JPERM(K)) IS FEASIBLE AND INCREASES THE         *****
C     *****   OBJECTIVE FUNCTION VALUE.                             *****
C
   40 LL=JPERM(K)
      LESS=0
      NLESS=0
      DO 41 I=1,M
        IF(SPRIME(I).GE.0.0) GO TO 41
        IF(A(I,LL))43,70,44
   43   LESS=1
        GO TO 41
   44   NLESS=2
   41   CONTINUE
      LJ=LESS+NLESS
      IF(LJ-2)60,50,70
   50 CONTINUE
C
C     *****   ALL ELEMENTS IN COLUMN JPERM(K) OF A ARE POSITIVE.   *****
C
      CALL PART5
      NBP5=NBP5+1
      IF(IMPROV.EQ.1)GO TO 30
      GO TO 70
C
C     *****   ALL ELEMENTS IN COLUMN JPERM(K) OF A ARE NEGATIVE.   *****
C
   60 CONTINUE
      CALL PART6
```

```
      NBR6=NBR6+1
      IF(IMPROV.EQ.1)GO TO 30
C
C     *****   CHECK TO SEE IF THERE ARE MORE K'S TO BE INVESTIGATED FOR    *****
C     *****   SIMULTANEOUS CHANGE.                                        *****
C
   70 IF(J.EQ.(K-1))GO TO 80
      K=K-1
      GO TO 40
   80 CONTINUE
      CALL PART7
      NBR7=NBR7+1
C
C     *****   CHECK TO SEE IF THERE ARE MORE J'S TO BE INVESTIGATED.   *****
C
      NT=N-1
      IF(NZERO.LT.NT)NT=NZERO
      IF(J.EQ.NT)GO TO 100
      J=J+1
      K=N
      GO TO 30
C
C     *****   CHECK TO SEE IF MODE 2 HAS FOUND A NEW SOLUTION.    *****
C     *****   IF SO RETURN TO MODE 1.                             *****
C
  100 DO 101 LWL=1,N
        IF(X(LWL).NE.XL(LWL))GO TO 20
  101   CONTINUE
  904 CONTINUE
C
C     *****   PRINT OUT DATA FROM PHASE 3.    *****
C
      WRITE(6,468)
      WRITE(6,476)NBR2,LSOLN2
      WRITE(6,464)NBR4
      WRITE(6,454)LSOLN4
      WRITE(6,465)NBR5
      WRITE(6,455)LSOLN5
      WRITE(6,466)NBR6
      WRITE(6,456)LSOLN6
      WRITE(6,467)NBR7
      WRITE(6,457)LSOLN7
      WRITE(6,483)
      WRITE(6,501)
      WRITE(6,500)(X(I),I=1,N)
      WRITE(6,502)
      WRITE(6,500)(S(I),I=1,M)
C
C     *****   PRINT OUT FINAL RESULTS.    *****
C
      ZZ=0.0
      TT=0.0
      T=0.0
      DO 102 J=1,N
        ZZ=ZZ+C(J)*X(J)
        TT=C(J)*XONE(J)+TT
        T=T+C(J)*XTWO(J)
  102   CONTINUE
      WRITE(6,5712)ZZ
      TEMP=CNEW4*(TT-T)
```

```
      WRITE(6,464)TEMP
      TEMP=CNORM0(TT-ZZ)
      WRITE(6,451)TEMP
      TEMP=CNORM0(ZZ-ZPRT2)
      WRITE(6,463)TEMP
      TEMP=CNORM0(ZZ-ZPH2)
      WRITE(6,450)TEMP
      REPEAT=REPEAT+1.0
C
C     *****   RPT IS THE NUMBER OF EXTRA SOLUTIONS TO BE OBTAINED.   *****
C     *****   CHECK TO SEE IF THIS REQUIREMENT HAS BEEN SATISFIED.   *****
C     *****   IF NOT FIND ANOTHER SOLUTION.                          *****
C
      IF(REPEAT.LE.RPT)GO TO 1969
C
C     *****   CHECK FOR PHASE 1, PHASE 2 COMBINATIONS    *****
C     *****   STILL TO BE TRIED.                         *****
C
 1999 GO TO (1900,1901),KPH
 1900 GO TO (165,166),KPH1
 1901 GO TO (167,1933),KPH1
 1933 CONTINUE
      IF(COUNT.LE.3.)GO TO 1932
      GO TO 1000
 2000 RETURN
      END
C
C
C
C
C
      SUBROUTINE SIMPLX(INFIX,A,B,TOL,PRM,KOUT,ERR,JH,X,P,Y,KB,E)
C
C
C     ********************************************************************
C     *      THE PURPOSE OF THIS CODE IS TO FIND THE SOLUTION OF THE LP PROBLEM:*
C     *                                                                  *
C     *                        MAXIMIZE    XZERO = CX                    *
C     *                  SUBJECT TO:                                     *
C     *                           AX . LE . B                            *
C     *                            X . GE . 0                            *
C     *                                                                  *
C     ********************************************************************
C
C
      DIMENSION INFIX(8),A(121,362),B(121),TOL(4),KOUT(7),ERR(8),ZZ(3),
     1JH(121),X(121),P(121),Y(121),KB(362),E(121,121),TERR(8),IOFIX(16)
      EQUIVALENCE (INFLAG,IOFIX(1)),(N,IOFIX(2)),
     1(ME,IOFIX(3)),(M,IOFIX(4)),(MF,IOFIX(5)),
     2(MC,IOFIX(6)),(NCUT,IOFIX(7)),(NVER,IOFIX(8)),
     3(K,IOFIX(9)),(ITER,IOFIX(10)),(INVC,IOFIX(11)),
     4(NUMVR,IOFIX(12)),(NUMPV,IOFIX(13)),
     5(INFS,IOFIX(14)),(JT,IOFIX(15)),(LA,IOFIX(15)),
     6(ZZ(1),TPIV),(ZZ(2),TZERO),(ZZ(3),TCOST)
C
C     *****   INITIALIZE VARIABLES.   *****
C
      DO 1340 I=1,8
        TERR(I)=0.0
        IOFIX(I+8)=0
```

```
      1340    IOFIX(I)=INFIX(I)
              DO 1308 I=1,3
      1308    ZZ(I)=TOL(I)
              TCOST=-ABS(TCOST)
              PRIX=PRP
              M2=M+2
              INFS=1
      C
      C       *****   CHECK FOR ILLEGAL INPUT.   *****
      C
              IF(N) 1304,1304,1371
      1371    IF(M-MF)1304,1304,1372
      1372    IF(MF-MC)1304,1304,1373
      1373    IF(MC)1304,1304,1374
      1374    IF(ME-M)1304,1375,1375
      1375    IF( MOD(INFLAG,4)-1)1400,1320,100
      C
      C       *****   RECORD INFORMATION ABOUT INITIAL BASIC FEASIBLE SOLUTION.     *****
      C       *****   (IT EXISTS DUE TO SET-UP IN MAIN PROGRAM.)                    *****
      C       *****      JH(I) = J    MEANS J IS THE (I-1)ST BASIC VARIABLE         *****
      C       *****                   INCLUDING SLACKS AND ARTIFICIALS.           *****
      C       *****      KB(J) = I    MEANS VARIABLE J IS BASIC IN ROW (I-1).       *****
      C       *****      KB(J) = 0    MEANS J IS NONBASIC.                          *****
      C       *****   (RECALL THAT THE COST COEFFICIENTS HAVE BEEN INSTALLED IN     *****
      C       *****   ROW 1 OF A.)                                                  *****
      C
      1400    DO 1401 I=1,M
      1401      JH(I)=0
      C
      C       *****   INITIALIZE KB AND JH TO REPRESENT THE INITIAL BASIC    *****
      C       *****   FEASIBLE SOLUTION.  THIS SOLUTION EXISTS DUE TO        *****
      C       *****   SET-UP IN MAIN PROGRAM.                                *****
      C
              DO 1402 J=1,N
                KB(J)=0
                KQ=0
                LQ=0
                DO 1403 L=MF,M
                  IF(A(L,J))1404,1403,1404
      1404        KQ=LQ+1
                  LQ=L
      1403        CONTINUE
      C
      C       *****   CHECK WHETHER VARIABLE J IS A CANDIDATE TO BE BASIC.   *****
      C
                IF(KQ-1)1402,1405,1402
      1405      IA=LQ
                IF(JH(IA))1402,1406,1402
      1406      IF(A(IA,J)*B(IA))1402,1407,1407
      C
      C
      C       *****   J IS A BASIC VARIABLE IN ROW (IA-1).    *****
      C       *****   SET JH AND KB ACCORDINGLY.              *****
      C
      C
      1407      JH(IA)=J
                KB(J)=IA
      1402      CONTINUE
      1320    CONTINUE
      1100    ASSIGN 1102 TO KPIV
```

```
720.            ASSIGN 1111 TO KJMY
721.            IF(LA)1121,1121,1122
722.      1121  INVC=0
723.      1122  NUMVR=NUMVR+1
724.            DO 1101 I=1,M
725.              DO 9 J=1,M
726.      9         E(I,J)=0.
727.      1101    CONTINUE
728.      C
729.      C       *****   INITIALIZE INVERSE AT IDENTITY AND STORE IN E.  ALSO    *****
730.      C       *****   INITIALIZE X, WHERE X(I) IS THE VALUE OF THE BASIC      *****
731.      C       *****   VARIABLE IN ROW (I-1).                                  *****
732.      C
733.            DO 1113 I=1,M
734.              E(I,I)=1.0
735.              X(I)=B(I)
736.      1113    CONTINUE
737.            DO 1110 I=MF,N
738.              IF(JH(I))1111,1110,1111
739.      1111    JH(I)=12345
740.      1110    CONTINUE
741.            INFS=1
742.            JT=1
743.      1     IF(KB(JT))600,1102,600
744.      C
745.      C       *****   FIND MAX OVER I OF ABS(Y(I)) = TY, STORE THE     *****
746.      C       *****   ARG MAX IN IR.                                   *****
747.      C
748.      1114  TY=0.
749.            DO 1104 I=MF,M
750.              IF(JH(I)-12345)1104,1105,1104
751.      1105    IF(ABS(Y(I))-TY)1104,1104,1106
752.      1106    IR=I
753.              TY=ABS(Y(I))
754.      1104    CONTINUE
755.            IF(TY-TPIV)1107,1108,1108
756.      1107  KB(JT)=0
757.            GO TO 1102
758.      C
759.      C       *****   VARIABLE JT BECOMES BASIC IN ROW (IR-1).    *****
760.      C
761.      1108  JH(IR)=JT
762.            KB(JT)=IR
763.            GO TO 900
764.      1102  JT=JT+1
765.            IF(JT.GT.N)GO TO 2
766.            GO TO 1
767.      2     DO 1109 I=1,M
768.              IF(JH(I)-12345)1109,1112,1109
769.      1112    JH(I)=0
770.      1109    CONTINUE
771.      100   ASSIGN 705 TO NDEL
772.            ASSIGN 1000 TO KJMY
773.            ASSIGN 221 TO KPIV
774.      C
775.      C       *****   CHECK FOR INFEASIBILITIES.  NEG = 1 MEANS    *****
776.      C       *****   X(I) . LT . 0 FOR SOME I.   JIN = 1 MEANS    *****
777.      C       *****   X(I) . LT . 0 FOR SOME I OR JH(I) = 0 FOR    *****
778.      C       *****   SOME I.                                      *****
779.      C
```

```
      1200  JIN=0
            NEG=0
            DO 1201 I=MF,M
              IF(ABS(X(I))-TZERO)1202,1203,1203
      1202    X(I)=0.0
              GO TO 1201
      1203    IF(X(I))1208,1201,1205
      1205    IF(JH(I))1201,1206,1201
      1208    NEG=1
      1206    JIN=1
      1201    CONTINUE
            IF(INFS-JIN)1320,500,200
C
C     *****   A FEASIBLE SOLUTION HAS BEEN FOUND.   *****
C
       200  INFS=0
       201  PMIX=0.0
       500  CONTINUE
C
C     *****   INITIALIZE PRICES.   *****
C
       502  DO 503 J=1,M
              P(J)=E(I,J)
       503    CONTINUE
            IF(INFS)501,599,501
C
C     *****   UPDATE PRICES.   *****
C
       501  DO 504 J=1,M
       504    P(J)=P(J)*PMIX
            DO 505 I=MF,M
              IF(X(I))506,507,507
       506    DO 508 J=1,M
                P(J)=P(J)+E(I,J)
       508      CONTINUE
              GO TO 505
       507    IF(JH(I)) 505,509,505
       509    DO 510 J=1,M
                P(J)=P(J)-E(I,J)
       510      CONTINUE
       505    CONTINUE
       599  CONTINUE
C
C     *****   FIND A COLUMN TO ENTER THE BASIS.   *****
C
       700  JT=0
            BB=TCOST
       701  JM=1
C
C     *****   SKIP COLUMNS ALREADY IN THE BASIS.   *****
C
       703  IF(KB(JM))702,300,702
C
C     *****   FIND THE MOST NEGATIVE PRICED OUT COLUMN.   *****
C     *****   STORE THE INDEX IN JT.                      *****
C
       705  IF(BB-BE)708,702,702
       708  BB=BE
            JT=JM
       702  JM=JM+1
```

```
      IF(JM.GT.N)GO TO 3
      GO TO 703
    3 IF(JT)203,203,600
C
C     *****   ALL COSTS ARE NON-NEGATIVE...K=3 OR 4.   *****
C
  203 K=3+INFS
      GO TO 257
C
C     *****   JT IS A BASIC VARIABLE.  STORE    *****
C     *****   (ROW J OF E) * (COLUMN JT OF A)   *****
C     *****   IN Y(J).                          *****
C
  600 DO 610 I=1,M
  610   Y(I)=0.
      DO 605 I=1,M
        IF(A(I,JT))601,602,601
  601   DO 606 J=1,M
  606     Y(J)=Y(J)+A(I,JT)*E(J,I)
        GO TO 605
  602   CONTINUE
  605   CONTINUE
  609 GO TO KJMY,(1000,1114,1392)
C
C     *****   FIND WHICH ROW JT (VIA Y) WILL BECOME BASIC IN.   *****
C
 1000 IR=0
      AA=0.0
      IA=0
C
C     *****   IF THERE ARE X(I) = 0, STORE MAX AMONG THESE I OF ABS(Y)   *****
C     *****   IN AA, STORE THE ARG MAX IN IR.                           *****
C
      DO 1050 I=MF,M
        IF(X(I))1050,1041,1050
 1041   YI=ABS(Y(I))
        IF(YI-TPIV)1050,1050,1042
 1042   IF(JM(I))1043,1044,1043
 1043   IF(IA)1050,1048,1050
 1048   IF(Y(I))1050,1050,1045
 1044   IF(IA)1045,1046,1045
 1045   IF(YI-AA)1050,1050,1047
 1046   IA=1
 1047   AA=YI
        IR=I
 1050   CONTINUE
      IF (IR) 1099,1301,1099
 1091 AA=1.0E+20
C
C     *****   THERE ARE NO X(I) = 0.  FIND THE MINIMUN PIVOT AMONG   *****
C     *****   THE POSITIVE EQUATIONS.  STORE THE MINIMUM VALUE IN    *****
C     *****   AA, WITH THE ARG MIN IN IR.                            *****
C
      DO 1010 IT=MF,M
        IF(Y(IT)-TPIV) 1010,1010,1002
 1002   IF(X(IT))1010,1010,1003
 1003   XY=X(IT)/Y(IT)
        IF(XY-AA)1004,1005,1010
 1005   IF(JM(IT))1010,1004,1010
 1004   AA=XY
```

```
900.                IR=IT
901.      1010      CONTINUE
902.                IF(NEG)1016,1099,1016
903.      C
904.      C         *****   FIND PIVOT AMONG NEGATIVE EQUATIONS, IN WHICH   *****
905.      C         *****   X/Y IS LESS THAN AA, THAT HAS THE LARGEST        *****
906.      C         *****   ABS(Y), STORE THE ARG MAX IN IR.                 *****
907.      C
908.      1016      BB=-TPIV
909.                DO 1030 I=MF,M
910.                  IF(X(I))1012,1030,1030
911.      1012        IF(Y(I)-BB)1022,1030,1030
912.      1022        IF(Y(I)*AA-X(I)) 1024,1024,1030
913.      1024        BB=Y(I)
914.                  IR=I
915.      1030      CONTINUE
916.      1099      CONTINUE
917.      206       IF(IR)207,207,210
918.      C
919.      C         *****   NO PIVOT ROW HAS BEEN FOUND FOR COLUMN JT.   *****
920.      C
921.      207       K=5
922.      257       IF(PMIX)201,400,201
923.      C
924.      C         *****   CHECK IF ITERATION LIMIT IS EXCEEDED.   *****
925.      C
926.      210       IF(ITER-NCUT) 900,160,160
927.      900       NUMPV=NUMPV+1
928.                YI=-Y(IR)
929.                Y(IR)=-1.
930.      C
931.      C         *****   TRANSFORM INVERSE TO CORRESPOND TO   *****
932.      C         *****   JT BEING BASIC IN ROW (IR-1).        *****
933.      C
934.      903       DO 904 L=1,M
935.                  IF(E(IR,L))905,914,905
936.      914         CONTINUE
937.                  GO TO 904
938.      905         XY=E(IR,L)/YI
939.                  E(IR,L)=0
940.                  DO 906 I=1,M
941.      906           E(I,L)=E(I,L)+XY*Y(I)
942.      904         CONTINUE
943.      C
944.      C         *****   TRANSFORM X TO CORRESPOND TO   *****
945.      C         *****   JT BEING BASIC IN ROW (IR-1).  *****
946.      C
947.                XY=X(IR)/YI
948.                X(IR)=0.
949.                DO 908 I=1,M
950.      908         X(I)=X(I)+XY*Y(I)
951.                Y(IR)=-YI
952.      999       GO TO KPIV,(221,1102)
953.      C
954.      C         *****   ADJUST JH AND KB TO MAKE JT BASIC IN ROW (IR-1).   *****
955.      C
956.      221       IA=JH(IR)
957.                IF(IA)213,213,214
958.      214       KB(IA)=0
959.      213       KB(JT)=IR
```

```
      JM(IR)=JT
      LA=0
      ITER=ITER+1
      INVC=INVC+1
C
C     *****   CHECK INVERSION FREQUENCY.   *****
C
      IF(INVC-NVER)1200,1320,1200
C
C     *****   TOO MANY ITERATIONS HAVE BEEN PERFORMED.   *****
C
160   K=4
C
C     *****   STORE AX - B AT Y.   *****
C
      ASSIGN 410 TO NDEL
      DO 401 I=1,M
401     Y(I)=-B(I)
      DO 402 I=1,M
        JA=JM(I)
        IF(JA) 403,402,403
403     DO 405 IT=1,M
          IF(A(IT,JA))415,405,415
415       Y(IT)=Y(IT)+X(I)*A(IT,JA)
405       CONTINUE
402     CONTINUE
C
C     *****   FIND SUM AND MAXIMUM OF ERRORS IN AX - B. STORE   *****
C     *****   IN TERR(1) AND TERR(2) RESPECTIVELY.              *****
C
      DO 481 I=1,M
        YI=Y(I)
        IF(JM(I)) 472,471,472
471     YI=YI+X(I)
472     TERR(LA+1)=TERR(LA+1)+ABS(YI)
        IF(ABS(TERR(LA+2))-ABS(YI))482,481,481
482     TERR(LA+2)=YI
481     CONTINUE
C
C     *****   STORE P TIMES BASIS AT DT.   *****
C
      I=1
4     JM=JM(I)
      IF(JM)300,411,300
C
C     *****   FIND SUM AND MAXIMUM OF ERRORS IN PA - D. STORE   *****
C     *****   IN TERR(3) AND TERR(4) RESPECTIVELY.              *****
C
410   TERR(LA+3)=TERR(LA+3)+ABS(DT)
      IF(ABS(TERR(LA+4)) - ABS(DT)) 413,411,411
413   TERR(LA+4)=DT
411   I=I+1
      IF(I.GT.M)GO TO 5
      GO TO 4
5     IF(LA) 193,191,193
191   LA=4
      IF(INFLAG-4)1320,193,193
193   IF(K-5)1392,194,1392
194   ASSIGN 1392 TO KJMY
      GO TO 600
```

```
1304  K=7
C
C     *****   SET EXIT VALUES FOR RETURN TO MAIN PROGRAM.   *****
C
1302  DO 1309 I=1,8
1309    ERR(I)=TERR(I)
      DO 1329 I=1,7
1329    KOUT(I)=IOFIX(I+8)
      RETURN
C
C     *****   TEST JM FOR ENTERRING THE BASIS.  PRICE OUT   *****
C     *****   JM AND STORE THIS VALUE IN DT.  (I.E. STORE   *****
C     *****   P * (COLUMN JM OF A) IN DT.)                  *****
C
303   DT=0.
301   DO 303 MM=1,M
        IF(A(MM,JM))304,303,304
304     DT=DT+P(MM)*A(MM,JM)
303     CONTINUE
399   GO TO NOEL,(410,705)
      END
C
C
C
C
C
      SUBROUTINE PHTWO1
C
C
C     ***************************************************************
C     *     THIS IS PHASE 2. CRITERION 1 (CORRESPONDING TO CRITERION 8 IN   *
C     *     REFERENCE 1 (P. 608).  THE 'INFEASIBILITY' OF A SOLUTION X IS   *
C     *                                                                     *
C     *          QMAX = MAX (COMPONENTS OF AX-B) .                          *
C     *                                                                     *
C     *     Q(I) MEASURES THE INFEASIBILITY IN CONSTRAINT I.                *
C     *                                                                     *
C     *     THE 'IMPROVEMENT' OBTAINED BY CHANGING THE VALUE OF A VARIABLE  *
C     *     X(J) IS                                                         *
C     *                                                                     *
C     *                  - (CHANGE IN QMAX) .                               *
C     ***************************************************************
C
C
      DIMENSION A(121,362),B(121),C(121),CAPR(121,121),CPERM(121),
     XDELX(121),DELTA(121),JPERM(121),LQ(121),Q(121),QSTAR(121),
     XS(121),SPRIME(121),X(121),XL(121),XONE(121),XTWO(121)
      COMMON A,ALPHA,B,C,CAPR,CPERM,DELX,I,IMPROV,DELTA,INVEST,J,
     XJHIGH,JJ,JPERM,K,KEY,LESS,LJ,LK,LL,LM,LML,LQ,LTEMP,M,METHOD,
     XNLESS,NZERO,PHI,Q,QMAX,QSTAR,S,SLPRIM,SLSUBK,SPRIME,SUM,T,
     XTEMP,TT,X,XL,XONE,XTWO,USUBK,ZM,ZZ,N,LSOLN2,LSOLN4,LSOLN5,
     XLSOLN6,LSOLN7
      IF(ZM.EQ.0.0)GO TO 1
C
C     *****     IDENTIFY POINT ON LINE SEGMENT JOINING XONE AND    *****
C     *****     XTWO THAT HAS BEEN REACHED.  STORE IT IN XL.       *****
C     *****     NOTICE THAT AS ALPHA INCREASES WE APPROACH XTWO    *****
C     *****     GOING ALONG SEGMENT JOINING XONE AND XTWO.         *****
C
      DO 1900 J=1,N
```

```
1080.      1000    XL(J)=XONE(J)+ALPHA*(XTWO(J)-XONE(J))
1081.            ZZ=.05
1082.            GO TO 10
1083.      1     ALPHA=0.0
1084.            ZZ=.05
1085.      2     TEMP=1.0-ALPHA
1086.      C
1087.      C     *****    IDENTIFY POINT ON LINE SEGMENT JOINING XONE AND    *****
1088.      C     *****    XTWO THAT HAS BEEN REACHED. STORE IT IN XL.        *****
1089.      C     *****    NOTICE THAT AS ALPHA INCREASES WE APPROACH XTWO    *****
1090.      C     *****    GOING ALONG SEGMENT JOINING XONE AND XTWO.         *****
1091.      C
1092.            DO 15 J=1,N
1093.      15      XL(J)=XONE(J)+ALPHA*(XTWO(J)-XONE(J))
1094.      C
1095.      C     *****    NOW OBTAIN NEAREST INTEGER POINT.  VARIABLES       *****
1096.      C     *****    THAT GO NEGATIVE ARE SET TO ZERO.                  *****
1097.      C
1098.            DO 820 J=1,N
1099.              IF(XL(J))821,821,822
1100.      821     X(J)=0.0
1101.              GO TO 820
1102.      822     X(J)=AINT(XL(J)+.5)
1103.      820     CONTINUE
1104.      C
1105.      C     *****    FIND QMAX, THE MEASURE OF INFEASIBILITY OF X.      *****
1106.      C
1107.            DO 19 I=1,M
1108.              T=0.0
1109.              DO 18 J=1,N
1110.      18        T=T+A(I,J)*X(J)
1111.      19      Q(I)=T-B(I)
1112.            QMAX=Q(1)
1113.            DO 20 I=2,M
1114.              IF(QMAX.GT.Q(I))GO TO 20
1115.              QMAX=Q(I)
1116.      20      CONTINUE
1117.      5     IF(QMAX.GT.0.0)GO TO 6
1118.            RETURN
1119.      C
1120.      C     *****    FIND DIRECTION OF FAVORABLE CHANGE FOR EACH        *****
1121.      C     *****    X(J) IF ONE EXISTS.  LESS AND NLESS INDICATE       *****
1122.      C     *****    NEGATIVE AND POSITIVE A(I,J) RESPECTIVELY.         *****
1123.      C
1124.      6     DO 22 J=1,N
1125.              LESS=0
1126.              NLESS=0
1127.              DO 23 I=1,M
1128.                IF(QMAX.GT.Q(I))GO TO 23
1129.                IF(A(I,J))27,26,28
1130.      27        IF(NLESS.EQ.1)GO TO 26
1131.                LESS=1
1132.                GO TO 23
1133.      28        IF(LESS.EQ.1)GO TO 26
1134.                NLESS=1
1135.                GO TO 23
1136.      26        DELX(J)=0.0
1137.                GO TO 22
1138.      23        CONTINUE
1139.              IF(LESS.EQ.1)GO TO 30
```

```
1140.                IF(X(J).LE.0.0)GO TO 17
1141.                DELX(J)=-1.0
1142.                GO TO 22
1143.       17       DELX(J)=0.0
1144.                GO TO 22
1145.       30       DELX(J)=1.0
1146.       22       CONTINUE
1147.       C
1148.       C     *****   IDENTIFY WHICH VARIABLE WHEN CHANGED WOULD GIVE   *****
1149.       C     *****   THE LARGEST 'IMPROVEMENT'.  STORE INDEX IN K.     *****
1150.       C
1151.             DO 42 J=1,N
1152.               IF(DELX(J))32,40,35
1153.       32      TT=Q(1)-A(1,J)
1154.               DO 36 I=2,M
1155.                 T=Q(I)-A(I,J)
1156.                 IF(T.GT.TT)TT=T
1157.       36        CONTINUE
1158.               QSTAR(J)=TT
1159.               GO TO 42
1160.       35      TT=Q(1)+A(1,J)
1161.               DO 21 I=2,M
1162.                 T=Q(I)+A(I,J)
1163.                 IF(T.GT.TT)TT=T
1164.       21        CONTINUE
1165.               QSTAR(J)=TT
1166.               GO TO 42
1167.       40      QSTAR(J)=QMAX
1168.       42      CONTINUE
1169.             K=0
1170.             T=-1.0E34
1171.             DO 45 J=1,N
1172.               IF(QMAX.LE.QSTAR(J))GO TO 45
1173.               TEMP=QMAX-QSTAR(J)
1174.               IF(TEMP.LE.T)GO TO 45
1175.               K=J
1176.               T=TEMP
1177.       45      CONTINUE
1178.             IF(K.EQ.0)GO TO 10
1179.       C
1180.       C     *****   CHANGE X(K) AND RECOMPUTE INFEASIBILITIES.   *****
1181.       C
1182.             X(K)=X(K)+DELX(K)
1183.             LJ=LJ+1
1184.             QMAX=QSTAR(K)
1185.             IF(DELX(K))54,5,46
1186.       54    DO 55 I=1,M
1187.               Q(I)=Q(I)-A(I,K)
1188.       55      CONTINUE
1189.             GO TO 5
1190.       46    DO 48 I=1,M
1191.               Q(I)=Q(I)+A(I,K)
1192.       48      CONTINUE
1193.             GO TO 5
1194.       C
1195.       C     *****   NO FURTHER REDUCTION IN INFEASIBILITY CAN BE ATTAINED.   *****
1196.       C     *****   MOVE TO A NEW POINT ON SEARCH SEGMENT.                   *****
1197.       C
1198.       10    IF(PH1.EQ.2.0)GO TO 340
1199.       C
```

```
C     *****   THIS IS PHASE 2, METHOD 1 (METHOD 3 IN REFERENCE  *****
C     *****   1 (P. 610).  INCREASE ALPHA BY THE MINIMUM        *****
C     *****   AMOUNT REQUIRED TO OBTAIN A DIFFERENT ROUNDED     *****
C     *****   SOLUTION THAN AT XL.                              *****
C
      DO 830 J=1,N
        TEMP=XTWO(J)-XONE(J)
        IF(TEMP)831,832,833
832     QSTAR(J)=10.0**34
        GO TO 830
831     IF(XL(J).LT.0.5)GO TO 832
        QSTAR(J)=-(XL(J)+.5001-AINT(XL(J)+.5))/TEMP
        GO TO 830
833     QSTAR(J)=(AINT(XL(J)+.5)+.5001-XL(J))/TEMP
830     CONTINUE
      QMAX=10.0**34
      DO 835 J=1,N
835     IF(QSTAR(J).LT.QMAX)QMAX=QSTAR(J)
      ALPHA=ALPHA+QMAX
      GO TO 841
C
C     *****   THIS IS PHASE 2, METHOD 2.  INCREASE ALPHA BY A   *****
C     *****   FIXED INCREMENT ZZ.                               *****
C
840   ALPHA=ALPHA+ZZ
841   CONTINUE
      LK=LK+1
      IF(ALPHA.LE.1.0)GO TO 2
C
C     *****   ALPHA .GT. 1. PHASE 2 HAS FAILED TO FIND A        *****
C     *****   FEASIBLE INTEGER SOLUTION.                        *****
C
      METHOD=0
      RETURN
      END
C
C
C
C
C
      SUBROUTINE PHTWO2
C
C
C     ****************************************************************
C     *     THIS IS PHASE 2, CRITERION 2 (CORRESPONDING TO CRITERION C IN      *
C     *     REFERENCE 1 (P. 608).  THE 'INFEASIBILITY' OF A SOLUTION X IS      *
C     *                                                                        *
C     *         QMAX = SUM OF THE POSITIVE COMPONENTS OF AX - B .              *
C     *                                                                        *
C     *     Q(I) MEASURES THE INFEASIBILITY IN CONSTRAINT I.                   *
C     *                                                                        *
C     *     THE 'IMPROVEMENT' OBTAINED BY CHANGING THE VALUE OF A VARIABLE     *
C     *     X(J) IS                                                            *
C     *                                                                        *
C     *           (C(J)*(CHANGE IN X(J)))/ - (CHANGE IN QMAX).                 *
C     ****************************************************************
C
C
      DIMENSION A(121,362),B(121),C(121),CAPR(121,121),CPERM(121),
     XDELX(121),DELTA(121),JPERM(121),LQ(121),Q(121),QSTAR(121),
```

```
      XS(121),SPRIME(121),X(121),XL(121),XONE(121),XTWO(121)
       COMMON A,ALPHA,B,C,CAPR,CPERM,DELX,I,IMPROV,DELTA,INVEST,J,
      XJHIGH,JJ,JPERM,K,KEY,LESS,LJ,LK,LL,LM,LML,LO,LTEMP,M,METHOD,
      XNLESS,NZERO,PHI,Q,QMAX,QSTAR,S,SLPRIM,SLSUBK,SPRIME,SUM,T,
      XTEMP,IT,X,XL,XONE,XTWO,JSUBK,ZM,ZZ,N,LSOLN2,LSOLN4,LSOLN5,
      XLSOLN6,LSOLN7
       IF(Z4.EQ.0.0)GO TO 1
C
C     *****   IDENTIFY POINT ON LINE SEGMENT JOINING XONE AND    *****
C     *****   XTWO THAT HAS BEEN REACHED. STORE IT IN XL.        *****
C     *****   NOTICE THAT AS ALPHA INCREASES WE APPROACH XTWO    *****
C     *****   GOING ALONG SEGMENT JOINING XONE AND XTWO.         *****
C
       DO 1900 J=1,N
1900     XL(J)=X(J)+ALPHA*(XTWO(J)-XONE(J))
       ZZ=.05
       GO TO 10
1      ALPHA=0.0
       ZZ=.05
2      TEMP=1.0-ALPHA
C
C     *****   IDENTIFY POINT ON LINE SEGMENT JOINING XONE AND    *****
C     *****   XTWO THAT HAS BEEN REACHED. STORE IT IN XL.        *****
C     *****   NOTICE THAT AS ALPHA INCREASES WE APPROACH XTWO    *****
C     *****   GOING ALONG SEGMENT JOINING XONE AND XTWO.         *****
C
       DO 15 J=1,N
15       XL(J)=XONE(J)+ALPHA*(XTWO(J)-XONE(J))
C
C     *****   NOW OBTAIN NEAREST INTEGER POINT.  VARIABLES       *****
C     *****   THAT GO NEGATIVE ARE SET TO ZERO.                  *****
C
       DO 820 J=1,N
         IF(XL(J))821,821,822
821      X(J)=0.0
         GO TO 820
822      X(J)=AINT(XL(J)+.5)
823      CONTINUE
C
C     *****   FIND QMAX, THE MEASURE OF INFEASIBILITY OF X.      *****
C
       DO 19 I=1,M
         T=0.0
         DO 18 J=1,N
           T=T+A(I,J)*X(J)
19       Q(I)=T-B(I)
       QMAX=0.0
       DO 530 I=1,M
         IF(Q(I).GT.0.0)QMAX=QMAX+Q(I)
530      CONTINUE
5      IF(QMAX.GT.0.0)GO TO 6
       RETURN
6      CONTINUE
C
C     *****   IDENTIFY WHICH VARIABLE WHEN CHANGED WOULD GIVE    *****
C     *****   THE LARGEST 'IMPROVEMENT'.  STORE INDEX IN K.      *****
C     *****   ALSO IDENTIFY DIRECTION OF FAVORABLE CHANGE FOR    *****
C     *****   EACH X(J).                                         *****
C
       LL=M
```

```
      KEY=0
      DO 535 J=1,N
        T=0.0
        TT=0.0
        DO 536 I=1,LL
          IF(Q(I))536,536,537
  537     T=T+A(I,J)
  536     CONTINUE
        IF(T)538,539,540
  538   DELX(J)=1.0
        GO TO 541
  540   IF(X(J).LE.0.0)GO TO 539
        DELX(J)=-1.0
        GO TO 542
  539   DELX(J)=0.0
        LQ(J)=0
        GO TO 535
  541   DO 543 I=1,LL
          T=Q(I)+A(I,J)
          IF(T.GT.0.0)TT=TT+T
  543     CONTINUE
        GO TO 545
  542   DO 544 I=1,LL
          T=Q(I)-A(I,J)
          IF(T.GT.0.0)TT=TT+T
  544     CONTINUE
  545   QSTAR(J)=TT
        IF(QSTAR(J).GT.0.0)GO TO 1112
C
C     *****   FEASIBILITY HAS BEEN ATTAINED.      *****
C
        X(J)=X(J)+DELX(J)
        RETURN
 1112   CONTINUE
        IF(TT-QMAX)546,547,547
  546   LQ(J)=1
        K=J
        KEY=KEY+1
        GO TO 535
  547   LQ(J)=0
  535   CONTINUE
      IF(KEY-1)10,9,47
   47 TEMP=-10**34
      DO 55 J=1,N
        IF(LQ(J).EQ.0)GO TO 55
        IF(((C(J)*DELX(J))/(QMAX-QSTAR(J))).LE.TEMP)GO TO 55
        JHIGH=J
        TEMP=(C(J)*DELX(J))/(QMAX-QSTAR(J))
   55   CONTINUE
      K=JHIGH
C
C     *****   CHANGE X(K) AND RECOMPUTE INFEASIBILITIES.   *****
C
    9 X(K)=X(K)+DELX(K)
      LJ=LJ+1
      T=DELX(K)
      DO 50 I=1,LL
   50   Q(I)=Q(I)+A(I,K)*T
      QMAX=QSTAR(K)
      GO TO 5
```

```
1380.      C
1381.      C     *****   NO FURTHER REDUCTION IN INFEASIBILITY CAN BE ATTAINED.   *****
1382.      C     *****   MOVE TO A NEW POINT OF SEARCH SEGMENT.                   *****
1383.      C
1384.      10    CONTINUE
1385.            IF(PHI.EQ.2.0)GO TO 840
1386.      C
1387.      C     *****   THIS IS PHASE 2, METHOD 1 (METHOD 3 IN REFERENCE  *****
1388.      C     *****   1 (P. 610).  INCREASE ALPHA BY THE MINIMUM        *****
1389.      C     *****   AMOUNT REQUIRED TO OBTAIN A DIFFERENT ROUNDED     *****
1390.      C     *****   SOLUTION THAN AT XL.                              *****
1391.      C
1392.            DO 830 J=1,N
1393.              TEMP=XT*O(J)-XONE(J)
1394.              IF(TEMP)831,832,833
1395.      832     QSTAR(J)=10.0**34
1396.              GO TO 830
1397.      831     IF(XL(J).LT.0.5)GO TO 832
1398.              QSTAR(J)=-(XL(J)+.5001-AINT(XL(J)+.5))/TEMP
1399.              GO TO 830
1400.      833     QSTAR(J)=(AINT(XL(J)+.5)+.5001-XL(J))/TEMP
1401.      830     CONTINUE
1402.            QMAX=10.0**34
1403.            DO 835 J=1,N
1404.      835     IF(QSTAR(J).LT.QMAX)QMAX=QSTAR(J)
1405.            ALPHA=ALPHA+QMAX
1406.            GO TO 841
1407.      C
1408.      C     *****   THIS IS PHASE 2, METHOD 2.  INCREASE ALPHA BY    *****
1409.      C     *****   A FIXED INCREMENT ZZ.                            *****
1410.      C
1411.      840   ALPHA=ALPHA+ZZ
1412.      841   CONTINUE
1413.            LK=LK+1
1414.            IF(ALPHA.LE.1.0)GO TO 2
1415.      C
1416.      C     *****   FEASIBLE INTEGER SOLUTION.   *****
1417.      C
1418.            METHOD=0
1419.            RETURN
1420.            END
1421.      C
1422.      C
1423.      C
1424.      C
1425.      C
1426.            SUBROUTINE PHI*03
1427.      C
1428.      C
1429.      C     ************************************************************************
1430.      C     *     THIS IS PHASE 2, CRITERION 3 (CORRESPONDING TO CRITERION A IN    *
1431.      C     *     REFERENCE 1 (P. 608).  THE 'INFEASIBILITY' OF A SOLUTION X IS    *
1432.      C     *                                                                      *
1433.      C     *         QMAX = SUM OF THE POSITIVE COMPONENTS OF AX - B .            *
1434.      C     *                                                                      *
1435.      C     *     Q(I) MEASURES THE INFEASIBILITY IN CONSTRAINT I.                 *
1436.      C     *                                                                      *
1437.      C     *     THE 'IMPROVEMENT' OBTAINED BY CHANGING THE VALUE OF A VARIABLE   *
1438.      C     *     X(J) IS                                                          *
1439.      C     *                                                                      *
```

```
C      *                    - (CHANGE IN QMAX) .                                    *
C      ************************************************************************
C
C
      DIMENSION A(121,362),B(121),C(121),CAPR(121,121),CPERM(121),
     XDELX(121),DELTA(121),JPERM(121),LQ(121),Q(121),QSTAR(121),
     XS(121),SPRIME(121),X(121),XL(121),XONE(121),XTWO(121)
      COMMON A,ALPHA,B,C,CAPR,CPERM,DELX,I,IMPROV,DELTA,INVEST,J,
     XJHIGH,JJ,JPERM,K,KEY,LESS,LJ,LK,LL,LM,LML,LQ,LTEMP,M,METHOD,
     XMLESS,NZERO,PHI,Q,QMAX,QSTAR,S,SLPRIM,SLSUBK,SPRIME,SUM,T,
     XTEMP,TT,X,XL,XONE,XTWO,USUBK,ZM,ZZ,N,LSOLN2,LSOLN4,LSOLN5,
     XLSOLN6,LSOLN7
      IF(ZM.EQ.0.0)GO TO 1
C
C     *****   IDENTIFY POINT ON LINE SEGMENT JOINING XONE AND    *****
C     *****   XTWO THAT HAS BEEN REACHED, STORE IT IN XL.        *****
C     *****   NOTICE THAT AS ALPHA INCREASES WE APPROACH XTWO    *****
C     *****   GOING ALONG SEGMENT JOINING XONE AND XTWO.         *****
C
      DO 1900 J=1,N
 1900   XL(J)=XONE(J)+ALPHA*(XTWO(J)-XONE(J))
      ZZ=.05
      GO TO 10
 1    ALPHA=0.0
      ZZ=.05
 2    TEMP=1.0-ALPHA
C
C     *****   IDENTIFY POINT ON LINE SEGMENT JOINING XONE AND    *****
C     *****   XTWO THAT HAS BEEN REACHED, STORE IT IN XL.        *****
C     *****   NOTICE THAT AS ALPHA INCREASES WE APPROACH XTWO    *****
C     *****   GOING ALONG SEGMENT JOINING XONE AND XTWO.         *****
C
      DO 15 J=1,N
 15     XL(J)=XONE(J)+ALPHA*(XTWO(J)-XONE(J))
C
C     *****   NOW OBTAIN NEAREST INTEGER POINT X.  VARIABLES     *****
C     *****   THAT GO NEGATIVE ARE SET TO ZERO.                  *****
C
      DO 820 J=1,N
        IF(XL(J))821,821,822
 821    X(J)=0.0
        GO TO 820
 822    X(J)=AINT(XL(J)+.5)
 820    CONTINUE
C
C     *****   FIND QMAX, THE MEASURE OF INFEASIBILITY OF X.     *****
C
      DO 19 I=1,M
        T=0.0
        DO 18 J=1,N
 18       T=T+A(I,J)*X(J)
 19     Q(I)=T-B(I)
      QMAX=0.0
      DO 530 I=1,M
        IF(Q(I).GT.0.0)QMAX=QMAX+Q(I)
 530    CONTINUE
 5    IF(QMAX.GT.0.0)GO TO 6
      RETURN
 6    CONTINUE
C
```

```
1500.      C      *****   IDENTIFY WHICH VARIABLE WHEN CHANGED WOULD    *****
1501.      C      *****   GIVE THE LARGEST 'IMPROVEMENT'.  STORE        *****
1502.      C      *****   INDEX IN KK IF FEASIBILITY IS ATTAINED, IN    *****
1503.      C      *****   K IF NOT.  ALSO IDENTIFY DIRECTION OF FAVOR-  *****
1504.      C      *****   ABLE CHANGE FOR EACH X(J).                    *****
1505.      C
1506.            LL=M
1507.            KEY=0
1508.            TL=-10**34
1509.            KK=0
1510.            DO 535 J=1,N
1511.              T=0.0
1512.              TT=0.0
1513.              DO 536 I=1,LL
1514.                IF(Q(I))536,536,537
1515.      537       T=T+A(I,J)
1516.      536       CONTINUE
1517.              TL1=ABS(T)
1518.              IF(T)538,539,540
1519.      538     DELX(J)=1.0
1520.              GO TO 541
1521.      540     IF(X(J).LE.0.0)GO TO 539
1522.              DELX(J)=-1.0
1523.              GO TO 542
1524.      539     DELX(J)=0.0
1525.              LQ(J)=0
1526.              GO TO 535
1527.      541     DO 543 I=1,LL
1528.                T=Q(I)+A(I,J)
1529.                IF(T.GT.0.0)TT=TT+T
1530.      543       CONTINUE
1531.              GO TO 545
1532.      542     DO 544 I=1,LL
1533.                T=Q(I)-A(I,J)
1534.                IF(T.GT.0.0)TT=TT+T
1535.      544       CONTINUE
1536.      545     QSTAR(J)=TT
1537.              IF(QSTAR(J).GT.0.0)GO TO 1112
1538.              IF(TL1.LE.TL)GO TO 535
1539.              TL=TL1
1540.              KK=J
1541.              GO TO 535
1542.      1112    CONTINUE
1543.              IF(TT-QMAX)546,547,547
1544.      546     LQ(J)=1
1545.              K=J
1546.              KEY=KEY+1
1547.              GO TO 535
1548.      547     LQ(J)=0
1549.      535     CONTINUE
1550.            IF(KK.EQ.0)GO TO 8
1551.      C
1552.      C     *****   FEASIBILITY HAS BEEN ATTAINED.   *****
1553.      C
1554.            X(KK)=X(KK)+DELX(KK)
1555.            RETURN
1556.      8     IF(KEY-1)10,9,47
1557.      47    TEMP=-10**34
1558.            DO 55 J=1,N
1559.              IF(LQ(J).EQ.0)GO TO 55
```

```
1560.                IF((QMAX-QSTAR(J)).LE.TEMP)GO TO 55
1561.                JHIGH=J
1562.                TEMP=QMAX-QSTAR(J)
1563.       55       CONTINUE
1564.       C
1565.       C        *****    CHANGE X(K) AND RECOMPUTE INFEASIBILITIES.    *****
1566.       C
1567.                K=JHIGH
1568.       9        X(K)=X(K)+DELX(K)
1569.                LJ=LJ+1
1570.                T=DELX(K)
1571.                DO 50 I=1,LL
1572.       50         Q(I)=Q(I)+A(I,K)*T
1573.                QMAX=QSTAR(K)
1574.                GO TO 5
1575.       10       CONTINUE
1576.       C
1577.       C        *****    NO FURTHER REDUCTION IN INFEASIBILITY CAN BE    *****
1578.       C        *****    ATTAINED.  MOVE TO A NEW POINT ON SEARCH        *****
1579.       C        *****    SEGMENT.                                        *****
1580.       C
1581.                IF(PHI.EQ.2.0)GO TO 840
1582.       C
1583.       C        *****    THIS IS PHASE 2, METHOD 1 (METHOD 3 IN REFERENCE  *****
1584.       C        *****    1 (P.610).  INCREASE ALPHA BY THE MINIMUM         *****
1585.       C        *****    AMOUNT REQUIRED TO OBTAIN A DIFFERENT ROUNDED     *****
1586.       C        *****    SOLUTION THAN AT XL.                              *****
1587.       C
1588.                DO 830 J=1,N
1589.                  TEMP=XTWO(J)-XONE(J)
1590.                  IF(TEMP)831,832,833
1591.       832        QSTAR(J)=10.0**34
1592.                  GO TO 830
1593.       831        IF(XL(J).LT.0.5)GO TO 832
1594.                  QSTAR(J)=-(XL(J)+.5001-AINT(XL(J)+.5))/TEMP
1595.                  GO TO 830
1596.       833        QSTAR(J)=(AINT(XL(J)+.5)+.5001-XL(J))/TEMP
1597.       830        CONTINUE
1598.                QMAX=10.0**34
1599.                DO 835 J=1,N
1600.       835        IF(QSTAR(J).LT.QMAX)QMAX=QSTAR(J)
1601.                ALPHA=ALPHA+QMAX
1602.                GO TO 841
1603.       C
1604.       C        *****    THIS IS PHASE 2, METHOD 2.  INCREASE ALPHA BY   *****
1605.       C        *****    A FIXED INCREMENT ZZ.                           *****
1606.       C
1607.       840      ALPHA=ALPHA+ZZ
1608.       841      CONTINUE
1609.                LK=LK+1
1610.                IF(ALPHA.LE.1.0)GO TO 2
1611.       C
1612.       C        *****    ALPHA .GT. 1, PHASE 2 HAS FAILED TO FIND A      *****
1613.       C        *****    FEASIBLE INTEGER SOLUTION.                      *****
1614.       C
1615.                METHOD=0
1616.                RETURN
1617.                END
1618.       C
1619.       C
```

```
      C
      C
      C
            SUBROUTINE PHTWO4
      C
      C
      C     ****************************************************************************
      C     *     THIS IS PHASE 2, CRITERION 4 (CORRESPONDING TO CRITERION D IN         *
      C     *     REFERENCE 1 (P. 608).  THE 'INFEASIBILITY' OF A SOLUTION X IS        *
      C     *                                                                          *
      C     *          QMAX = MAX (COMPONENTS OF AX - B) .                             *
      C     *                                                                          *
      C     *     Q(I) MEASURES IN INFEASIBILITY IN CONSTRAINT I.                      *
      C     *                                                                          *
      C     *     THE 'IMPROVEMENT' OBTAINED BY CHANGING THE VALUE OF A VARIABLE       *
      C     *     X(J) IS                                                              *
      C     *                                                                          *
      C     *                  (C(J)*(CHANGE IN X(J)))/ - (CHANGE IN QMAX).            *
      C     ****************************************************************************
      C
      C
            DIMENSION A(121,362),B(121),C(121),CAPR(121,121),CPERM(121),
           XDELX(121),DELTA(121),JPERM(121),LQ(121),Q(121),QSTAR(121),
           XS(121),SPRIME(121),X(121),XL(121),XONE(121),XTWO(121)
            COMMON A,ALPHA,B,C,CAPR,CPERM,DELX,I,IMPROV,DELTA,INVEST,J,
           XJHIGH,JJ,JPERM,K,KEY,LESS,LJ,LK,LL,LM,LML,LQ,LTEMP,M,METHOD,
           XNLESS,NZERO,PHI,Q,QMAX,QSTAR,S,SLPRIM,SLSUBK,SPRIME,SUM,T,
           XTEMP,TT,X,XL,XONE,XTWO,USUBK,ZM,ZZ,N,LSOLN2,LSOLN4,LSOLN5,
           XLSOLN6,LSOLN7
            IF(ZM.EQ.0.0)GO TO 1
      C
      C     *****   IDENTIFY POINT ON LINE SEGMENT JOINING XONE AND    *****
      C     *****   XTWO THAT HAS BEEN REACHED. STORE IT IN XL.        *****
      C
            DO 1900 J=1,N
      1900    XL(J)=XONE(J)+ALPHA*(XTWO(J)-XONE(J))
            ZZ=.05
            GO TO 10
      1     ALPHA=0.0
            ZZ=.05
      2     TEMP=1.0-ALPHA
      C
      C     *****   IDENTIFY POINT ON LINE SEGMENT JOINING XONE AND    *****
      C     *****   XTWO THAT HAS BEEN REACHED. STORE IT IN XL.        *****
      C
            DO 15 J=1,N
      15      XL(J)=XONE(J)+ALPHA*(XTWO(J)-XONE(J))
      C
      C     *****   NOW OBTAIN NEAREST INTEGER POINT X.  VARIABLES     *****
      C     *****   THAT GO NEGATIVE ARE SET TO ZERO.                  *****
      C
            DO 820 J=1,N
              IF(XL(J))821,821,822
      821     X(J)=0.0
              GO TO 820
      822     X(J)=AINT(XL(J)+.5)
      820     CONTINUE
      C
      C     *****   FIND QMAX, THE MEASURE OF INFEASIBILITY OF X.     *****
      C
```

```
      DO 10 I=1,4
        T=0.0
        DO 10 J=1,4
   10     T=T+A(I,J)*X(J)
   11   Q(I)=T-B(I)
      QMAX=Q(1)
      DO 20 I=2,4
        IF(QMAX.GT.Q(I))GO TO 20
        QMAX=Q(I)
   20   CONTINUE
    5 IF(QMAX.GT.0.0) GO TO 6
      RETURN
C
C     *****   FIND DIRECTION OF FAVORABLE CHANGE FOR EACH X(J)   *****
C     *****   IF ONE EXISTS.  LESS AND NLESS INDICATE NEGATIVE   *****
C     *****   AND POSITIVE A(I,J) RESPECTIVELY.                  *****
C
    6 DO 22 J=1,N
        LESS=0
        NLESS=0
        DO 23 I=1,4
          IF(QMAX.GT.Q(I))GO TO 23
          IF(A(I,J))27,26,28
   27     IF(NLESS.EQ.1)GO TO 26
          LESS=1
          GO TO 23
   28     IF(LESS.EQ.1)GO TO 26
          NLESS=1
          GO TO 23
   26     DELX(J)=0.0
          GO TO 22
   23     CONTINUE
        IF(LESS.EQ.1)GO TO 30
        IF(X(J).LE.0.0)GO TO 17
        DELX(J)=-1.0
        GO TO 22
   17   DELX(J)=0.0
        GO TO 22
   30   DELX(J)=1.0
   22   CONTINUE
C
C     *****   IDENTIFY WHICH VARIABLE WHEN CHANGED WOULD GIVE   *****
C     *****   THE LARGEST 'IMPROVEMENT'.  STORE INDEX IN K.     *****
C
      DO 42 J=1,N
        IF(DELX(J))32,40,35
   32   TT=Q(1)-A(1,J)
        DO 36 I=2,4
          T=Q(I)-A(I,J)
          IF(T.GT.TT)TT=T
   36     CONTINUE
        GSTAR(J)=TT
        GO TO 1125
   35   TT=Q(1)+A(1,J)
        DO 21 I=2,4
          T=Q(I)+A(I,J)
          IF(T.GT.TT)TT=T
   21     CONTINUE
        GSTAR(J)=TT
        GO TO 1125
```

```
      40        QSTAR(J)=QMAX
      1125      CONTINUE
                IF(QSTAR(J).GT.0.0)GO TO 42
                X(J)=X(J)+DELX(J)
                RETURN
      42        CONTINUE
              K=0
              T=-10.0**34
              DO 45 J=1,N
                IF(QMAX.LE.QSTAR(J))GO TO 45
                TEMP=(C(J)*DELX(J))/(QMAX-QSTAR(J))
                IF(TEMP.LE.T)GO TO 45
                K=J
                T=TEMP
      45        CONTINUE
              IF(K.EQ.0)GO TO 10
      C
      C       *****   CHANGE X(K) AND RECOMPUTE INFEASIBILITIES.     *****
      C
              X(K)=X(K)+DELX(K)
              LJ=LJ+1
              QMAX=QSTAR(K)
              IF(DELX(K))54,5,46
      54      DO 55 I=1,M
                Q(I)=Q(I)-A(I,K)
      55        CONTINUE
              GO TO 5
      46      DO 48 I=1,M
                Q(I)=Q(I)+A(I,K)
      48        CONTINUE
              GO TO 5
      C
      C       *****   NO FURTHER REDUCTION IN INFEASIBILITY CAN BE ATTAINED.   *****
      C       *****   MOVE TO A NEW POINT ON SEARCH SEGMENT.                   *****
      C
      10      IF(PH1.EQ.2.0)GO TO 840
      C
      C       *****   THIS IS PHASE 2, METHOD 1 (METHOD 3 IN REFERENCE  *****
      C       *****   1 (P. 610).  INCREASE ALPHA BY THE MINIMUM        *****
      C       *****   AMOUNT REQUIRED TO OBTAIN A DIFFERENT ROUNDED     *****
      C       *****   SOLUTION THAN AT XL.                              *****
      C
              DO 830 J=1,N
                TEMP=XTWO(J)-XONE(J)
                IF(TEMP)831,832,833
      832       QSTAR(J)=10.0**34
                GO TO 830
      831       IF(XL(J).LT.0.5)GO TO 832
                QSTAR(J)=-(XL(J)+.5001-AINT(XL(J)+.5))/TEMP
                GO TO 830
      833       QSTAR(J)=(AINT(XL(J)+.5)+.5001-XL(J))/TEMP
      830       CONTINUE
              QMAX=10.0**34
              DO 835 J=1,N
      835       IF(QSTAR(J).LT.QMAX)QMAX=QSTAR(J)
              ALPHA=ALPHA+QMAX
              GO TO 841
      C
      C       *****   THIS IS PHASE 2, METHOD 2.  INCREASE ALPHA BY A   *****
      C       *****   FIXED INCREMENT ZZ.                               *****
```

```
1900.       C
1901.  440  ALPHA=ALPHA+ZZ
1902.  441  CONTINUE
1903.       LK=LK+1
1904.       IF(ALPHA.LE.1.0)GO TO 2
1905.       C
1906.       C     *****   ALPHA . GT . 1, PHASE 2 HAS FAILED TO FIND A   *****
1907.       C     *****   FEASIBLE INTEGER SOLUTION.                     *****
1908.       C
1909.       METHOD=0
1910.       RETURN
1911.       END
1912.       C
1913.       CC
1914.       CC
1915.       CC
1916.       C
1917.       SUBROUTINE PART1
1918.       C
1919.       C
1920.       C     ***************************************************************************
1921.       C     *      INITIALIZE THE FIRST MODE OF SEARCH.  THIS MODE CONSISTS OF        *
1922.       C     *      SUCCESSIVELY INCREASING OR DECREASING SOME VARIALBE BY ONE AS      *
1923.       C     *      LONG AS EACH RESULTING SOLUTION IS BOTH FEASIBLE AND BETTER THAN   *
1924.       C     *      THE PRECEDING ONE.                                                 *
1925.       C     ***************************************************************************
1926.       C
1927.       C
1928.         DIMENSION A(121,362),B(121),C(121),CAPR(121,121),CPERM(121),
1929.      XDELX(121),DELTA(121),JPERM(121),LQ(121),Q(121),QSTAR(121),
1930.      XS(121),SPRIME(121),X(121),XL(121),XONE(121),XTWO(121)
1931.         COMMON A,ALPHA,B,C,CAPR,CPERM,DELX,I,IMPROV,DELTA,INVEST,J,
1932.      XJHIGH,JJ,JPERM,K,KEY,LESS,LJ,LK,LL,LM,LML,LQ,LTEMP,M,METHOD,
1933.      XNLESS,NZERO,PHI,Q,QMAX,QSTAR,S,SLPRIM,SLSUBK,SPRIME,SUM,T,
1934.      XTEMP,TT,X,XL,XONE,XTWO,YSUBK,ZM,ZZ,N,LSOLN2,LSOLN4,LSOLN5,
1935.      XLSOLN6,LSOLN7
1936.       C
1937.       C     *****   IDENTIFY (JPERM(1).....JPERM(N)) A PERMUTATION OF (1.....N)  *****
1938.       C     *****   SUCH THAT                                                    *****
1939.       C     *****     ABS(C(JPERM(1))) . GE .    ...     . GE . ABS(C(JPERM(N))) *****
1940.       C     *****   AND SET                                                      *****
1941.       C     *****                CPERM(I) = C(JPERM(I))  FOR I = 1.....N         *****
1942.       C
1943.         DO 101 J=1,N
1944.           JPERM(J)=J
1945.  101      CPERM(J)=C(J)
1946.         LML=N-1
1947.         DO 102 I=1,LML
1948.           TEMP=ABS(CPERM(I))
1949.           LM=I+1
1950.           DO 103 J=LM,N
1951.             IF(ABS(CPERM(J)).LE.TEMP)GO TO 103
1952.             JJ=JPERM(I)
1953.             JPERM(I)=JPERM(J)
1954.             JPERM(J)=JJ
1955.             TEMP=ABS(CPERM(J))
1956.             T=CPERM(I)
1957.             CPERM(I)=CPERM(J)
1958.             CPERM(J)=T
1959.  103        CONTINUE
```

```
1860.      102    CONTINUE
1861.             C
1862.             C     *****   COMPUTE THE NUMBER OF C(J)'S . NE . 0 AND STORE    *****
1863.             C     *****   THIS VALUE IN NZERO.                               *****
1864.             C
1865.             DO 104 J=1,N
1866.      104      IF(ABS(CPERM(J)).GT.0.0)NZERO=J
1867.             C
1868.             C     *****   STORE THE SIGN OF CPERM(I) IN DELTA(I).    *****
1869.             C
1870.             DO 105 J=1,NZERO
1871.               LL=JPERM(J)
1872.               IF(CPERM(J))106,105,107
1873.      106      DELTA(LL)=-1.0
1874.               GO TO 105
1875.      107      DELTA(LL)=1.0
1876.      105      CONTINUE
1877.             C
1878.             C     *****   LET S(I) = SLACK IN CONSTRAINT I.   *****
1879.             C
1880.             DO 109 I=1,M
1881.               TEMP=0.0
1882.               DO 110 J=1,N
1883.      110        TEMP=TEMP+A(I,J)*X(J)
1884.      109      S(I)=B(I)-TEMP
1885.             RETURN
1886.             END
1887.             C
1888.             C
1889.             C
1890.             C
1891.             C
1892.             SUBROUTINE PART2
1893.             C
1894.             C
1895.             C     ************************************************************************
1896.             C     *      THIS IS THE MAIN PART OF THE FIRST MODE OF SEARCH.  DETERMINE HOW  *
1897.             C     *      MUCH EACH OF THE VARIABLES CAN BE CHANGED IN THE FAVORABLE         *
1898.             C     *      DIRECTION BEFORE BECOMING INFEASIBLE.  THEN FIND WHICH VARIABLE    *
1899.             C     *      MAXIMIZES THE RATE OF INCREASE OF THE OBJECTIVE FUNCTION WITH      *
1900.             C     *      RESPECT TO THE PERCENTAGE DECREASE IN SLACK FOR THE VARIABLE.      *
1901.             C     *      STORE THE INDEX IN K.                                             *
1902.             C     ************************************************************************
1903.             C
1904.             C
1905.               DIMENSION A(121,362),B(121),C(121),CAPR(121,121),CPERM(121),
1906.            XDELX(121),DELTA(121),JPERM(121),LQ(121),Q(121),QSTAR(121),
1907.            XS(121),SPRIME(121),X(121),XL(121),XONE(121),XTWO(121)
1908.               COMMON A,ALPHA,B,C,CAPR,CPERM,DELX,I,IMPROV,DELTA,INVEST,J,
1909.            XJHIGH,JJ,JPERM,K,KEY,LESS,LJ,LK,LL,LM,LML,LQ,LTEMP,M,METHOD,
1910.            XNLESS,NZERO,PH1,Q,QMAX,QSTAR,S,SLPRIM,SLSUBK,SPRIME,SUM,T,
1911.            XTEMP,TT,X,XL,XONE,XTWO,XSUBK,ZM,ZZ,N,LSOLN2,LSOLN4,LSOLN5,
1912.            XLSOLN6,LSOLN7
1913.      111     K=1
1914.              TT=1.0
1915.              DO 95 J=1,N
1916.                IF(C(J))525,95,113
1917.      525       IF(X(J))95,95,113
1918.      113       TEMP=10.0**12
1919.                DO 118 I=1,M
```

```
                IF(C(J)*A(I,J))118,113,117
  117           T=S(I)/ABS(A(I,J))
                IF(METHOD.EQ.2)GO TO 942
C
C     *****   THIS IS PHASE 3, PART 2, METHOD 1.  THE INTEGER          *****
C     *****   RESTRICTION IS USED TO COMPUTE HOW MUCH A VARIABLE CAN   *****
C     *****   CHANGE AND REMAIN FEASIBLE.                              *****
C
                IF(T)120,122,121
  120           IF(T.EQ.AINT(T))GO TO 122
                T=AINT(T)-1.0
                GO TO 122
  121           T=AINT(T)
                GO TO 122
C
C     *****   THIS IS PHASE 3, PART 2, METHOD 2.  THE INTEGER  *****
C     *****   RESTRICTION IS IGNORED IN COMPUTING HOW MUCH A   *****
C     *****   VARAIBLE CAN CHANGE AND REMAIN FEASIBLE.         *****
C
  942           IF(T-1.0)941,122,122
  941           T=0.0
  122           IF(T.LT.TEMP)TEMP=T
  118           CONTINUE
              IF(TT.GT.ABS(C(J))*TEMP)GO TO 95
              TT=ABS(C(J))*TEMP
              K=J
   95         CONTINUE
              IF(TT)130,127,130
  127         RETURN
C
C     *****   CHANGE X(K) AND RECOMPUTE SLACKS.   *****
C
  130         CONTINUE
              LSOLN2=LSOLN2+1
              IF(C(K))131,131,132
  131         X(K)=X(K)-1.0
              DO 133 I=1,M
  133           S(I)=S(I)+A(I,K)
              GO TO 111
  132         X(K)=X(K)+1.0
              DO 137 I=1,M
  137           S(I)=S(I)-A(I,K)
              GO TO 111
              END
C
CC
CC
CC
C
              SUBROUTINE PART3
C
C
C     ***********************************************************************
C     *      INITIALIZE THE SECOND MODE OF SEARCH.  THIS MODE SEARCHES FOR  *
C     *      BETTER SOLUTIONS THAT CAN BE OBTAINED BY CHANGING TWO VARIABLES*
C     *      SIMULTANEOUSLY.  FOR EACH PAIR OF VARIABLES X(JPERM(J)) AND    *
C     *      X(JPERM(K)) THE ONLY CHANGES CONSIDERED IN THE VARIABLE        *
C     *      X(JPERM(J)) WITH LARGER ABSOLUTE VALUE OF OBJECTIVE COEFFICIENT*
C     *      IS TO ADD OR SUBTRACT ONE.  SINCE THE ONLY CHANGES THAT NEED BE*
C     *      CONSIDERED IN THE OTHER VARIABLE ARE THOSE THAT STILL EFFECT A *
```

```
C      *     NET INCREASE IN OBJECTIVE FUNCTION VALUE.  THE LOWER BOUND ON SJC4 *
C      *     CHANGES IS STORED IN CAPR(J,K);  THE UPPER BOUND IN CAPR(K,J).     *
C      ***************************************************************************
C
C
      DIMENSION A(121,362),B(121),C(121),CAPR(121,121),CPERM(121),
     XDELX(121),DELTA(121),JPERM(121),LQ(121),Q(121),QSTAR(121),
     XS(121),SPRIME(121),X(121),XL(121),XONE(121),XTWO(121)
      COMMON A,ALPHA,B,C,CAPR,CPERM,DELX,I,IMPROV,DELTA,INVEST,J,
     XJHIGH,JJ,JPERM,K,KEY,LESS,LJ,LK,LL,LM,LML,LQ,LTEMP,M,METHOD,
     XNLESS,NZERO,PHI,Q,QMAX,QSTAR,S,SLPRIM,SLSUBK,SPRIME,SUM,T,
     XTEMP,TT,X,XL,XONE,XTWO,USUBK,ZM,ZZ,N,LSOLN2,LSOLN4,LSOLN5,
     XLSOLN6,LSOLN7
      LM=NZERO-1
      DO 147 J=1,LM
        LL=J+1
        LJ=JPERM(J)
        DO 141 K=LL,NZERO
          LK=JPERM(K)
          TEMP=ABS(C(LJ)/C(LK))
          IF(TEMP-1.0)142,142,143
142       CAPR(J,K)=AINT(TEMP-1.0)
          GO TO 144
143       IF(AINT(TEMP).EQ.TEMP)GO TO 142
          CAPR(J,K)=AINT(TEMP-1.0)+1.0
144       IF(TEMP+1.0)145,146,146
146       CAPR(K,J)=AINT(TEMP+1.0)
          GO TO 141
145       IF(AINT(TEMP).EQ.TEMP)GO TO 146
          CAPR(K,J)=AINT(TEMP+1.0)-1.0
141       CONTINUE
147     CONTINUE
      RETURN
      END
C
C
C
C
C
      SUBROUTINE PART4
C
C
C     ***************************************************************************
C     *     THE FIRST CHANGE IN X(JPERM(J)) CONSIDERED IS THAT ONE WHICH WOULD *
C     *     INCREASE THE OBJECTIVE FUNCTION VALUE.  BEFORE CHECKING FOR        *
C     *     SIMULTANEOUS CHANGES IN X(JPERM(K)) TEST FOR FEASIBILITY.          *
C     ***************************************************************************
C
C
      DIMENSION A(121,362),B(121),C(121),CAPR(121,121),CPERM(121),
     XDELX(121),DELTA(121),JPERM(121),LQ(121),Q(121),QSTAR(121),
     XS(121),SPRIME(121),X(121),XL(121),XONE(121),XTWO(121)
      COMMON A,ALPHA,B,C,CAPR,CPERM,DELX,I,IMPROV,DELTA,INVEST,J,
     XJHIGH,JJ,JPERM,K,KEY,LESS,LJ,LK,LL,LM,LML,LQ,LTEMP,M,METHOD,
     XNLESS,NZERO,PHI,Q,QMAX,QSTAR,S,SLPRIM,SLSUBK,SPRIME,SUM,T,
     XTEMP,TT,X,XL,XONE,XTWO,USUBK,ZM,ZZ,N,LSOLN2,LSOLN4,LSOLN5,
     XLSOLN6,LSOLN7
      J=J
      LL=JPERM(J)
150   CONTINUE
```

```
      IF(X(LL)+DELTA(LL))151,152,152
C
C     *****   DO NOT INVESTIGATE CHANGE IN X(JPERM(J)) IN FAVORABLE    *****
C     *****   DIRECTION - IT WILL GO NEGATIVE.                         *****
C
151   INVEST=0
      RETURN
152   CONTINUE
      KEY=0
      DO 161 I=1,M
        SPRIME(I)=S(I)-DELTA(LL)*A(I,LL)
        IF(SPRIME(I))163,161,161
163     KEY=1
161     CONTINUE
      IF(KEY.EQ.1)GO TO 170
C
C     *****   THE CHANGE IN X(JPERM(J)) IS FEASIBLE SO    *****
C     *****   CHANGE VALUE AND RECOMPUTE INFEASIBILITIES.*****
C
      X(LL)=X(LL)+DELTA(LL)
      LSOLN4=LSOLN4+1
      DO 172 I=1,M
172     S(I)=SPRIME(I)
      GO TO 150
C
C     *****   INVESTIGATE CHANGE IN X(JPERM(J)) IN FAVORABLE DIRECTION.   *****
C
170   INVEST=1
      RETURN
      END
C
C
C
C
C
      SUBROUTINE PARTS
C
C
C     ***************************************************************************
C     *      INVESTIGATE A SIMULTANEOUS CHANGE IN A VARIABLE X(JPERM(K)) WHICH  *
C     *      IS FEASIBLE AND YIELDS A NET INCREASE IN THE OBJECTIVE FUNCTION    *
C     *      VALUE.  IN THIS CASE THE ONLY WAY OF RESTORING FEASIBILITY IS TO   *
C     *      DECREASE X(JPERM(K)) SUFFICIENTLY.                                 *
C     ***************************************************************************
C
C
      DIMENSION A(121,362),B(121),C(121),CAPR(121,121),CPERM(121),
     XDELX(121),DELTA(121),JPERM(121),LQ(121),Q(121),QSTAR(121),
     XS(121),SPRIME(121),X(121),XL(121),XONE(121),XTWO(121)
      COMMON A,ALPHA,B,C,CAPR,CPERM,DELX,I,IMPROV,DELTA,INVEST,J,
     XJHIGH,JJ,JPERM,K,KEY,LESS,LJ,LK,LL,LM,LML,LQ,LTEMP,M,METHOD,
     XNLESS,NZERO,PHI,G,QMAX,QSTAR,S,SLPRIM,SLSUBK,SPRIME,SUM,T,
     XTEMP,TT,X,XL,XONE,XTWO,USUBK,ZM,ZZ,N,LSOLN2,LSOLNA,LSOLN5,
     XLSOLN6,LSOLN7
      J=J
      K=K
      IF(CPERM(K))192,192,191
191   LL=JPERM(K)
      SLSUBK=-AMIN1(X(LL),CAPR(J,K))
      GO TO 199
```

```
2100.      192   LL=JPERM(K)
2101.            SLSUBK=-X(LL)
2102.      199   USUBK=10.0**12
2103.            LL=JPERM(K)
2104.            DO 206 I=1,M
2105.              IF(SPRIME(I).GE.0.0)GO TO 206
2106.              TEMP=SPRIME(I)/A(I,LL)
2107.              IF(TEMP.EQ.AINT(TEMP))GO TO 207
2108.              TEMP=AINT(TEMP)-1.0
2109.      207     IF(TEMP.LT.USUBK)USUBK=TEMP
2110.      206     CONTINUE
2111.            IF(SLSUBK-USUBK)220,220,24000
2112.      C
2113.      C     *****   THERE IS NO IMPROVED SOLUTION THAT CAN BE    *****
2114.      C     *****   ATTAINED BY CHANGING X(JPERM(J)) IN THE      *****
2115.      C     *****   FAVORABLE DIRECTION USING X(JPERM(K)).       *****
2116.      C
2117.      24000 IMPROV=0
2118.            RETURN
2119.      C
2120.      C     *****   CHECK IF SMALLEST DECREASE IN X(JPERM(K))    *****
2121.      C     *****   WHICH YIELDS FEASIBILITY ALSO BRINGS A       *****
2122.      C     *****   STRICT INCREASE IN XZERO.                    *****
2123.      C
2124.      220   LL=JPERM(K)
2125.            SLPRIM=-10.0**12
2126.            DO 221 I=1,M
2127.              IF(A(I,LL))224,221,221
2128.      224     TEMP=SPRIME(I)/A(I,LL)
2129.              IF(TEMP)223,226,227
2130.      223     TEMP=AINT(TEMP)
2131.              GO TO 226
2132.      227     IF(TEMP.EQ.AINT(TEMP))GO TO 226
2133.              TEMP=AINT(TEMP)+1.0
2134.      226     IF(TEMP.GT.SLPRIM)SLPRIM=TEMP
2135.      221     CONTINUE
2136.            IF(SLPRIM.GT.SLSUBK)SLSUBK=SLPRIM
2137.            IF(SLSUBK-USUBK)250,250,24000
2138.      C
2139.      C     *****   FEASIBLE CHANGES EXIST.  CHOOSE THE ONE THAT    *****
2140.      C     *****   YIELDS THE LARGEST INCREASE IN XZERO FOR THE    *****
2141.      C     *****   NEW SOLUTION.  UPDATE X(JPERM(J)), X(JPERM(K))  *****
2142.      C     *****   AND SLACKS.                                     *****
2143.      C
2144.      250   LL=JPERM(K)
2145.            LSOLNS=LSOLNS+1
2146.            IF(C(LL))251,251,252
2147.      251   LM=JPERM(J)
2148.            X(LM)=X(LM)+DELTA(LM)
2149.            X(LL)=X(LL)+SLSUBK
2150.            DO 253 I=1,M
2151.      253     S(I)=SPRIME(I)-SLSUBK*A(I,LL)
2152.      209   IMPROV=1
2153.            RETURN
2154.      252   X(LL)=X(LL)+USUBK
2155.            DO 254 I=1,M
2156.      254     S(I)=SPRIME(I)-USUBK*A(I,LL)
2157.            LL=JPERM(J)
2158.            X(LL)=X(LL)+DELTA(LL)
2159.            GO TO 209
```

```
2160.                 END
2161.       C
2162.       C
2163.       C
2164.       C
2165.       C
2166.                 SUBROUTINE PART6
2167.       C
2168.       C
2169.       C     ****************************************************************************
2170.       C     *      INVESTIGATE A SIMULTANEOUS CHANGE IN VARIABLE X(JPERM(K)) WHICH     *
2171.       C     *      IS FEASIBLE AND YIELDS A NET INCREASE IN THE OBJECTIVE FUNCTION     *
2172.       C     *      VALUE.  IN THIS CASE THE ONLY WAY OF RESTORING FEASIBILITY IS TO    *
2173.       C     *      INCREASE X(JPERM(K)) SUFFICIENTLY.                                  *
2174.       C     ****************************************************************************
2175.       C
2176.       C
2177.                 DIMENSION A(121,362),B(121),C(121),CAPR(121,121),CPERM(121),
2178.                XDELX(121),DELTA(121),JPERM(121),LQ(121),Q(121),QSTAR(121),
2179.                XS(121),SPRIME(121),X(121),XL(121),XONE(121),XTWO(121)
2180.                 COMMON A,ALPHA,B,C,CAPR,CPERM,DELX,I,IMPROV,DELTA,INVEST,J,
2181.                XJHIGH,JJ,JPERM,K,KEY,LESS,LJ,LK,LL,LM,LML,LQ,LTEMP,M,METHOD,
2182.                XNLESS,NZERO,PHI,Q,QMAX,QSTAR,S,SLPRIM,SLSUBK,SPRIME,SUM,T,
2183.                XTEMP,TT,X,XL,XONE,XTWO,USUBK,ZM,ZZ,N,LSOLN2,LSOLN4,LSOLN5,
2184.                XLSOLN6,LSOLN7
2185.                 J=J
2186.                 K=K
2187.                 IF(CPERM(K))260,261,261
2188.       260       USUBK=CAPR(J,K)
2189.                 GO TO 262
2190.       261       USUBK=10.0**12
2191.       262       CONTINUE
2192.                 LL=JPERM(K)
2193.                 TEMP=-10.0**12
2194.                 DO 276 I=1,M
2195.                   IF(SPRIME(I).GE.0.0)GO TO 276
2196.                   T=SPRIME(I)/A(I,LL)
2197.                   IF(AINT(T).EQ.T)GO TO 279
2198.                   T=AINT(T)+1.0
2199.       279         IF(TEMP-T)280,276,276
2200.       280         TEMP=T
2201.       276         CONTINUE
2202.                 SLSUBK=TEMP
2203.                 IF(SLSUBK-USUBK)300,300,24000
2204.       C
2205.       C     *****       THERE IS NO IMPROVED SOLUTION THAT CAN BE   *****
2206.       C     *****       ATTAINED BY CHANGING X(JPERM(J)) IN THE     *****
2207.       C     *****       FAVORABLE DIRECTION USING X(JPERM(K)).      *****
2208.       C
2209.       24000 IMPROV=0
2210.                 RETURN
2211.       C
2212.       C     *****    CHECK IF SMALLEST INCREASE IN X(JPERM(K))     *****
2213.       C     *****    WHICH YIELDS FEASIBILITY ALSO BRINGS A        *****
2214.       C     *****    STRICT INCREASE IN XZERO.                     *****
2215.       C
2216.       300       LL=JPERM(K)
2217.                 TEMP=10.0**12
2218.                 DO 307 I=1,M
2219.                   IF(A(I,LL))307,307,304
```

```
      T=SPRIME(I)/A(I,LL)
      IF(T)309,311,310
310   T=AINT(T)
      GO TO 311
309   IF(T.EQ.AINT(T))GO TO 311
      T=AINT(T)-1.0
311   IF(TEMP-T)307,307,312
312   TEMP=T
307   CONTINUE
      IF(TEMP.LT.USUBK)USUBK=TEMP
      IF(SLSUBK.LE.USUBK)USUBK=TEMP
      IF(SLSUBK.LE.USUBK)GO TO 250
      GO TO 24000
C
C     *****   FEASIBLE CHANGES EXIST.  CHOOSE THE ONE    *****
C     *****   THAT YIELDS THE LARGEST INCREASE IN XZERO *****
C     *****   FOR THE NEW SOLUTION.  UPDATE X(JPERM(J)),*****
C     *****   X(JPERM(K)) AND SLACKS.                   *****
C
250   LL=JPERM(K)
      LSOLN6=LSOLN6+1
      IF(C(LL))251,251,252
251   LM=JPERM(J)
      X(LM)=X(LM)+DELTA(LM)
      X(LL)=X(LL)+SLSUBK
      DO 253 I=1,M
253     S(I)=SPRIME(I)-SLSUBK*A(I,LL)
209   IMPROV=1
      RETURN
252   X(LL)=X(LL)+USUBK
      DO 254 I=1,M
254     S(I)=SPRIME(I)-USUBK*A(I,LL)
      LL=JPERM(J)
      X(LL)=X(LL)+DELTA(LL)
      GO TO 209
      END
C
C
C
C
C
      SUBROUTINE PART7
C
C
C     ********************************************************************************
C     *      INVESTIGATE THE POSSIBILITY OF CHANGING X(JPERM(J)) IN THE              *
C     *      UNFAVORABLE DIRECTION.  THE ONLY CHANGE CONSIDERED IN X(JPERM(K))       *
C     *      IS THE SMALLEST INTEGER CHANGE THAT YIELDS A STRICT INCREASE IN         *
C     *      XZERO.  CHECK FOR ALL K IN J+1,....,NZERO.                              *
C     ********************************************************************************
C
C
      DIMENSION A(121,362),B(121),C(121),CAPR(121,121),CPERM(121),
     XDELX(121),DELTA(121),JPERM(121),LQ(121),Q(121),QSTAR(121),
     XS(121),SPRIME(121),X(121),XL(121),XONE(121),XTWO(121)
      COMMON A,ALPHA,B,C,CAPR,CPERM,DELX,I,IMPROV,DELTA,INVEST,J,
     XJHIGH,JJ,JPERM,K,KEY,LESS,LJ,LK,LL,LM,LML,LQ,LTEMP,M,METHOD,
     XNLESS,NZERO,PHI,Q,QMAX,QSTAR,S,SLPRIM,SLSUBK,SPRIME,SUM,T,
     XTEMP,TT,X,XL,XONE,XTWO,USUBK,ZM,ZZ,N,LSOLN2,LSOLN4,LSOLN5,
     XLSOLN6,LSOLN7
```

```
2280.                J=J
2281.                K=K
2282.          C
2283.          C     ***** CHANGE X(JPERM(J)) IN UNFAVORABLE DIRECTION.  ADJUST SLACKS.  *****
2284.          C
2285.          25000 LL=JPERM(J)
2286.                IF(X(LL)-DELTA(LL))30000,330,330
2287.          30000 RETURN
2288.          330   DO 331 I=1,M
2289.          331     SPRIME(I)=S(I)+DELTA(LL)*A(I,LL)
2290.          335   IF(K-NZERO)340,340,30000
2291.          C
2292.          C     *****   CHECK TO SEE IF (X(JPERM(K)) + DELTA(JPERM(K))) . GE . 0. *****
2293.          C
2294.          340   LL=JPERM(K)
2295.                IF(X(LL)+DELTA(LL)*CAPR(K,J))341,350,350
2296.          341   K=K+1
2297.                GO TO 335
2298.          C
2299.          C     *****   CHECK TO SEE IF CONSTRAINTS ARE SATISFIED     *****
2300.          C     *****   BY (X(JPERM(K)) + DELTA(JPERM(K))).           *****
2301.          C
2302.          350   LL=JPERM(K)
2303.                DO 351 I=1,M
2304.                  IF(SPRIME(I)-DELTA(LL)*CAPR(K,J)*A(I,LL))352,351,351
2305.          352     K=K+1
2306.                  GO TO 335
2307.          351     CONTINUE
2308.          C
2309.          C     *****   A NEW FEASIBLE SOLUTION WITH BETTER OBJECTIVE   *****
2310.          C     *****   FUNCTION VALUE HAS BEEN FOUND.  ADJUST         *****
2311.          C     *****   X(JPERM(K)), X(JPERM(J)) AND SLACKS.           *****
2312.          C
2313.                LL=JPERM(J)
2314.                LSOLN7=LSOLN7+1
2315.                X(LL)=X(LL)-DELTA(LL)
2316.                LL=JPERM(K)
2317.                X(LL)=X(LL)+DELTA(LL)*CAPR(K,J)
2318.                DO 361 I=1,M
2319.          361     S(I)=SPRIME(I)-DELTA(LL)*CAPR(K,J)*A(I,LL)
2320.                GO TO 25000
2321.                END
```

APPENDIX II

SAMPLE INPUT FOR COMPUTER PROGRAM HEUR

```
//GO.SYSIN DD *
TEST PROBLEM
   15    16
  0.0
   1    0    1    1    0    1    0    1
 22.   40.   33. -36.   -1.   43.   56.    8.    6. -38.   12.    9. -16.   47.   33.
  5.
 19.   10.   40.   13.   26.   28.   38.   50.   17.   36.   29. -37.   23. -33.   46.
  5.
 17.   53. -16.   34. -30.   47.   13. -39. -25. -17.   42.   27.   50.   39.    3.
  5.
  4. -36. -31. -21.   34.   35. -16.   37.   11. -12. -15.   35. -32. -11. -33.
  5.
 16.    4. -23.   15.   10. -40.   27.   25. -14. -21. -40. -39.   55. -16.   30.
  5.
 -2. -22. -26.   22.   44. -33. -10.   58. -11.   47.   55. -24.    5. -30.   28.
  5.
-13.    9.   16. -12.   27.   15.   16. -23.   26.   31.   10.   22.   35. -30. -31.
  5.
-27.   30.   53.    5. -34. -14.    0.    0.   39.   18.   38. -36.   59.   51. -33.
  5.
-28.   12. -12. -25. -24. -34.   58. -27.   38. -30.   11.   29.    2.   40.   51.
  5.
-38.   25. -40.   50.   -3.    9.    5. -36.   21.   28.   29. -28.   56.   16.   35.
  5.
-24. -17.    0.   49. -35.   24.   34.   33. -11.   34.   42.   49.   56.    4.    3.
  5.
  8. -18.   13.   -5.   34.   20.   46.   36.   53. -16.   14.   29.   31.    5.   -9.
  5.
 58.   21.   38.   -6.   35.   -7.   41.   34.   48.    8. -37. -31.   15.   48.   55.
  5.
-32. -20.   27.   27.   25. -28.   16.   -8.   43.   59. -11.    4.   14.    3.   42.
  5.
 26.   17. -32. -21.   47. -37. -15.   20. -22.   10.    5.   34.   49. -27. -29.
  5.
834. 853. 979. 822. 813. 752. 822. 588. 911. 915. 979. 584. 810. 720. 590.
 58.    6.   60.   63.   -6.   71.   56. -14.   53.   43. -10.    3. -10.   -4.   52.
 40.
/*
//
```

APPENDIX III

SAMPLE OUTPUT FOR COMPUTER PROGRAM HEUR

```
      TEST PROBLEM
THE CONSTRAINT MATRIX A(I,J) IS:
   22.00    40.00    33.00   -36.00    -1.00    43.00    56.00     8.00     6.00   -38.00    12.00     9.00   -16.00    47.00    33.00
    5.00
   19.00    10.00    49.00    17.00    26.00    28.00    38.00    50.00    17.00    35.00    29.00   -37.00    23.00   -33.00    46.00
    5.00
   17.00    53.00   -16.00    34.00   -30.00    47.00    13.00   -39.00   -25.00   -17.00    42.00    27.00    50.00    39.00     3.00
    5.00
    4.00   -36.00   -31.00   -21.00    34.00    35.00   -15.00    37.00    11.00   -12.00   -15.00    35.00   -32.00   -11.00   -33.00
    5.00
   16.00     4.00   -23.00    15.00    10.00   -40.00    27.00    25.00   -14.00   -21.00   -40.00   -39.00    56.00   -16.00    30.00
    5.00
   -2.00   -22.00   -26.00    22.00    44.00   -33.00   -10.00    59.00   -11.00    47.00    55.00   -24.00     5.00   -30.00    28.00
    5.00
  -33.00     8.00    16.00   -12.00    27.00    15.00    16.00   -23.00    26.00    31.00    10.00    22.00    35.00   -30.00   -31.00
    5.00
  -27.00    30.00    53.00     5.00   -36.00   -14.00     0.0      0.0     39.00    18.00    38.00   -36.00    59.00    51.00   -33.00
    5.00
  -28.00    12.00   -12.00   -25.00   -24.00   -34.00    58.00   -27.00    38.00   -30.00    11.00    29.00     2.00    40.00    51.00
    5.00
  -38.00    25.00   -40.00    50.00    -3.00     9.00     5.00   -36.00    21.00    28.00    29.00   -28.00    56.00    16.00    35.00
    5.00
  -24.00   -17.00     0.0     49.00   -35.00    24.00    34.00    33.00   -11.00    34.00    42.00    49.00    56.00     4.00     3.00
    5.00
    8.00   -19.00    13.00    -5.00    36.00    20.00    46.00    36.00    53.00   -16.00    14.00    29.00    31.00     5.00    -9.00
    5.00
   58.00    21.00    39.00    -6.00    35.00    -7.00    41.00    34.00    48.00     8.00   -37.00   -31.00    15.00    48.00    55.00
    5.00
  -32.00   -20.00    27.00    27.00    25.00   -28.00    16.00    -8.00    43.00    59.00   -11.00     4.00    14.00     3.00    42.00
    5.00
   24.00    17.00   -32.00   -21.00    47.00   -37.00   -15.00    20.00   -22.00    10.00     5.00    34.00    49.00   -27.00   -29.00
    5.00
THE RIGHT HAND SIDE B(I) IS:
  834.00   853.00   979.00   822.00   812.00   752.00   822.00   588.00   911.00   915.00   979.00   584.00   810.00   720.00   590.00
THE COST COEFFICIENTS C(J) ARE:
   98.00     6.00    60.00    63.00    -6.00    71.00    56.00   -14.00    53.00    43.00   -10.00     3.00   -10.00    -4.00    52.00
   60.00
THE NORMALIZATION FACTOR FOR THE OBJECTIVE FUNCTION IS       0.01.
THE SIMPLEX METHOD GIVES US:
 THE LP SOLUTION NONE IS:
      2.81693      1.25460          0.0          6.63173          0.0          0.48790
          0.0          0.0          0.0          0.31753          0.0          0.0
          0.0          0.0          0.75636    123.86716
 FOLLOWING, IN ORDER OF INCREASING J, IS THE ROW X(J) IS BASIC IN:
       13       9        0        3        0       12        0        0        0        1        0        0        0        0        2
       14
 FOLLOWING, IN ORDER OF INCREASING I, IS THE ITH BASIC VARIABLE, INCLUDING SLACKS:
       10       15        4       20       21       18       23        2       25       17       27        6        1       16       31
 THE NORMALIZED INVERSE OF THE OPTIMAL BASIS IS:
          0.0          0.0          6.33003          0.0          0.0         -3.82294
          0.0    -0.09148           0.0         -8.41045          0.0         -2.24632
    -1.60377       9.19545          0.0          0.0          0.0         -2.06989
          0.0          0.0         -1.40126          0.0    -0.76006          0.0
      2.51634          0.0          0.45438          1.70634    -0.45927          0.0
          0.0          0.0         -0.43273          0.0          0.0          5.94474
          0.0    -1.19333           0.0          3.29456          0.0         -0.36586
    -1.41647       -5.29791         0.0          0.0          0.0         -1.45657
      1.00000          0.0          1.32734          0.0          0.02374          0.0
      2.25151          0.0         -1.05993          0.62613    -2.39565          0.0
```

```
      0.0          0.0          3.83045      0.0          1.00000      -3.42990
      0.0         -0.39525      0.0         -5.04976      0.0          -1.02417
     -1.41106      5.78087      0.0          0.0          1.00000      -2.22357
      0.0          0.0          1.79950      0.0          0.38091       0.0
      2.44416      0.0          0.30303     -0.00015     -3.36713       0.0
      0.0          0.0         -2.41590      0.0          0.0           3.06543
      1.00000     -0.97333      0.0          3.52693      0.0           0.00958
      0.62111     -4.66284      0.0          0.0          0.0           0.14462
      0.0          0.0         -1.38443      0.0          1.33398       0.0
      0.29554      0.0         -1.14435      1.05787      0.09891       0.0
      0.0          0.0          4.70723      0.0          0.0          -0.10822
      0.0         -0.90063      1.00000     -5.17720      0.0          -1.50617
     -2.22791      3.91342      0.0          1.00000      0.0           0.38365
      0.0          0.0          2.70460      0.0         -0.38344       0.0
     -0.23687      0.0         -0.80698     -1.43725     -1.08842       0.0
      0.0          0.0         -1.68813      0.0          0.0          -0.20241
      0.0          0.46843      0.0          1.20498      1.00000      -0.02652
      1.05218     -1.52327      0.0          0.0          0.0           5.39465
      0.0          0.0         -5.14769      0.0         -1.21517       0.0
     -6.08729      0.0         -0.31908     -1.46839      7.85910       0.0
      0.0          0.0          2.40270      0.0          0.0           2.04762
      0.0         -0.70377      0.0         -2.63566      0.0          -0.87693
     -0.25549      0.23492      0.0          0.0          0.0          -8.80472
      0.0          0.0          3.51965      0.0          7.93484       0.0
     10.54063      0.0         12.68879      6.61352     -8.15562       0.0
      0.0          0.0          0.41573      0.0          0.0          -1.03597
      0.0         -1.21444      0.0          0.10630      0.0          -0.04325
     -0.56556      0.96135      1.00000
 PHASE 1. METHOD 1.   PHASE 2. METHOD 2.
 PHASE 2. CRITERION 1.
 THE CURRENT VALUE OF ALPHA IS       0.0 .
 THE PHASE 1 SOLUTION XTWO IS:
    2.54     0.89     0.0      7.29     0.0     -1.27     0.0      0.0      0.0     -2.14     0.0      0.0      0.0      0.0      0.77
  114.74
AT THE END OF PHASE 2 WE HAVE THE FOLLOWING DATA:
  THE VALUE OF ALPHA WHERE THE FEASIBLE SOLUTION WAS FOUND =        0.0    .
  THE NUMBER OF POINTS MOVED TO ON THE SEARCH LINE =                  0.
  THE NUMBER OF TRIAL SOLUTIONS BEFORE FINDING THE FEASIBLE SOLU      5.
  THE SOLUTION FROM PHASE 2 IS:
    3.00     1.00     0.0      7.00     0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0
  120.00
FOR THE FIRST TIME THROUGH MODE 1 WE HAVE THE FOLLOWING DATA:
  THE NUMBER OF SOLUTION CHANGES =                                    0.
  THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE =     0.0    .
THE FOLLOWING OCCURRED PRIOR TO THE TERMINATION OF PHASE 3:
  DURING PHASE 3 WE ENTERED MODE 1   2 TIMES AND HAD   0 SOLUTION CHANGES.
  THE NUMBER OF TIMES THROUGH PART 4 =                               31.
  THE NUMBER OF SOLUTION CHANGES DURING PART 4 =                      0.
  THE NUMBER OF TIMES THROUGH PART 5 =                               60.
  THE NUMBER OF SOLUTION CHANGES DURING PART 5 =                      0.
  THE NUMBER OF TIMES THROUGH PART 6 =                               11.
  THE NUMBER OF SOLUTION CHANGES DURING PART 6 =                      1.
  THE NUMBER OF TIMES THROUGH PART 7 =                               30.
  THE NUMBER OF SOLUTION CHANGES DURING PART 7 =                      0.
AT THE END OF THE HEURISTIC PROCEDURE WE HAVE THE FOLLOWING RESULTS:
  THE FINAL APPROXIMATE INTEGER SOLUTION IS:
    3.00     1.00     0.0      7.00     0.0      1.00     0.0      0.0      0.0      1.00     0.0      0.0      0.0      0.0      0.0
  120.00
  THE FEASIBILITY TEST SLACKS ARE:
    3.09     0.25     0.05     3.58     1.05     0.10     3.78     0.00     5.07     0.14     0.53     0.09     0.40     0.15     0.62
```

```
 THE OBJECTIVE FUNCTION VALUE OF THE FINAL SOLUTION =                                    7935.00.
 THE NORMALIZED DIFFERENCE BETWEEN THE XONE AND XTWO OBJECTIVE FUNCTION VALUES =          4.3132.
 THE NORMALIZED DIFFERENCE BETWEEN THE OPTIMAL LP AND INTEGER OBJECTIVE FUNCTION VALUES =  0.9916.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE SINCE THE FIRST TIME THROUGH MODE  0.6521.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE IN PHASE 3 =                 0.6521.
PHASE 1. METHOD 2.  PHASE 2. METHOD 2.
PHASE 2. CRITERION 1.
THE CURRENT VALUE OF ALPHA IS       0.0 .
THE PHASE 1 SOLUTION XTWO IS:
    2.53     0.72     0.0      5.93     0.0      1.79     0.0      0.0      0.0      1.18     0.0      0.0      0.0      0.0      0.77
   91.67
AT THE END OF PHASE 2 WE HAVE THE FOLLOWING DATA:
 THE VALUE OF ALPHA WHERE THE FEASIBLE SOLUTION WAS FOUND =       0.0    .
 THE NUMBER OF POINTS MOVED TO ON THE SEARCH LINE =                    0.
 THE NUMBER OF TRIAL SOLUTIONS BEFORE FINDING THE FEASIBLE SOLU        5.
 THE SOLUTION FROM PHASE 2 IS:
    3.00     1.00     0.0      7.00     0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0
  120.00
FOR THE FIRST TIME THROUGH MODE 1 WE HAVE THE FOLLOWING DATA:
 THE NUMBER OF SOLUTION CHANGES =                                      0.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE =     0.0    .
THE FOLLOWING OCCURRED PRIOR TO THE TERMINATION OF PHASE 3:
 DURING PHASE 3 WE ENTERED MODE 1    2 TIMES AND HAD    0 SOLUTION CHANGES.
 THE NUMBER OF TIMES THROUGH PART 4 =                  31.
 THE NUMBER OF SOLUTION CHANGES DURING PART 4 =         0.
 THE NUMBER OF TIMES THROUGH PART 5 =                  60.
 THE NUMBER OF SOLUTION CHANGES DURING PART 5 =         0.
 THE NUMBER OF TIMES THROUGH PART 6 =                  11.
 THE NUMBER OF SOLUTION CHANGES DURING PART 6 =         1.
 THE NUMBER OF TIMES THROUGH PART 7 =                  30.
 THE NUMBER OF SOLUTION CHANGES DURING PART 7 =         0.
AT THE END OF THE HEURISTIC PROCEDURE WE HAVE THE FOLLOWING RESULTS:
 THE FINAL APPROXIMATE INTEGER SOLUTION IS:
    3.00     1.00     0.0      7.00     0.0      1.00     0.0      0.0      0.0      1.00     0.0      0.0      0.0      0.0      0.0
  120.00
 THE FEASIBILITY TEST SLACKS ARE:
    3.08     0.08     0.08     3.58     1.08     0.10     3.78     0.00     5.07     0.14     0.53     0.09     0.40     0.15     0.62
 THE OBJECTIVE FUNCTION VALUE OF THE FINAL SOLUTION =                                    7935.00.
 THE NORMALIZED DIFFERENCE BETWEEN THE XONE AND XTWO OBJECTIVE FUNCTION VALUES =         10.6700.
 THE NORMALIZED DIFFERENCE BETWEEN THE OPTIMAL LP AND INTEGER OBJECTIVE FUNCTION VALUES =  0.9916.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE SINCE THE FIRST TIME THROUGH MODE  0.6521.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE IN PHASE 3 =                 0.6521.
PHASE 1. METHOD 1.  PHASE 2. METHOD 2.
PHASE 2. CRITERION 3.
THE CURRENT VALUE OF ALPHA IS       0.0 .
THE PHASE 1 SOLUTION XTWO IS:
    2.54     0.99     0.0      7.29     0.0     -1.27     0.0      0.0      0.0     -2.14     0.0      0.0      0.0      0.0      0.77
  114.74
AT THE END OF PHASE 2 WE HAVE THE FOLLOWING DATA:
 THE VALUE OF ALPHA WHERE THE FEASIBLE SOLUTION WAS FOUND =       0.4000.
 THE NUMBER OF POINTS MOVED TO ON THE SEARCH LINE =                    8.
 THE NUMBER OF TRIAL SOLUTIONS BEFORE FINDING THE FEASIBLE SOLU        8.
 THE SOLUTION FROM PHASE 2 IS:
    3.00     1.00     0.0      7.00     0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0
  120.00
FOR THE FIRST TIME THROUGH MODE 1 WE HAVE THE FOLLOWING DATA:
 THE NUMBER OF SOLUTION CHANGES =                                      0.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE =     0.0    .
THE FOLLOWING OCCURRED PRIOR TO THE TERMINATION OF PHASE 3:
 DURING PHASE 3 WE ENTERED MODE 1    2 TIMES AND HAD    0 SOLUTION CHANGES.
```

```
 THE NUMBER OF TIMES THROUGH PART 4 =                    31.
 THE NUMBER OF SOLUTION CHANGES DURING PART 4 =           0.
 THE NUMBER OF TIMES THROUGH PART 5 =                    50.
 THE NUMBER OF SOLUTION CHANGES DURING PART 5 =           0.
 THE NUMBER OF TIMES THROUGH PART 6 =                    11.
 THE NUMBER OF SOLUTION CHANGES DURING PART 6 =           1.
 THE NUMBER OF TIMES THROUGH PART 7 =                    30.
 THE NUMBER OF SOLUTION CHANGES DURING PART 7 =           0.
AT THE END OF THE HEURISTIC PROCEDURE WE HAVE THE FOLLOWING RESULTS:
 THE FINAL APPROXIMATE INTEGER SOLUTION IS:
   3.00    1.00    0.0     7.00    0.0     1.00    0.0     0.0     0.0     1.00    0.0     0.0     0.0     0.0     0.0
  120.00
 THE FEASIBILITY TEST SLACKS ARE:
   3.08    0.25    0.05    3.58    1.05    0.10    3.78    0.00    5.07    0.14    0.53    0.09    0.40    0.15    0.62
 THE OBJECTIVE FUNCTION VALUE OF THE FINAL SOLUTION =                                                    7935.00.
 THE NORMALIZED DIFFERENCE BETWEEN THE XONE AND XTWO OBJECTIVE FUNCTION VALUES =                         4.3102.
 THE NORMALIZED DIFFERENCE BETWEEN THE OPTIMAL LP AND INTEGER OBJECTIVE FUNCTION VALUES =                0.9916.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE SINCE THE FIRST TIME THROUGH MODE            0.6521.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE IN PHASE 3 =                                 0.6521.
PHASE 1. METHOD 2.  PHASE 2. METHOD 2.
PHASE 2. CRITERION 3.
THE CURRENT VALUE OF ALPHA IS       0.0 .
THE PHASE 1 SOLUTION XTWO IS:
   2.53    0.72    0.0     5.93    0.0     1.79    0.0     0.0     0.0     1.18    0.0     0.0     0.0     0.0     0.77
  91.67
AT THE END OF PHASE 2 WE HAVE THE FOLLOWING DATA:
 THE VALUE OF ALPHA WHERE THE FEASIBLE SOLUTION WAS FOUND =        0.1500.
 THE NUMBER OF POINTS MOVED TO ON THE SEARCH LINE =                    3.
 THE NUMBER OF TRIAL SOLUTIONS BEFORE FINDING THE FEASIBLE SOLU        3.
 THE SOLUTION FROM PHASE 2 IS:
   3.00    1.00    0.0     7.00    0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     1.00
  119.00
FOR THE FIRST TIME THROUGH MODE 1 WE HAVE THE FOLLOWING DATA:
 THE NUMBER OF SOLUTION CHANGES =                                      0.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE =    0.0    .
THE FOLLOWING OCCURRED PRIOR TO THE TERMINATION OF PHASE 3:
 DURING PHASE 3 WE ENTERED MODE 1   2 TIMES AND HAD   0 SOLUTION CHANGES.
 THE NUMBER OF TIMES THROUGH PART 4 =                    32.
 THE NUMBER OF SOLUTION CHANGES DURING PART 4 =           1.
 THE NUMBER OF TIMES THROUGH PART 5 =                   100.
 THE NUMBER OF SOLUTION CHANGES DURING PART 5 =           2.
 THE NUMBER OF TIMES THROUGH PART 6 =                    26.
 THE NUMBER OF SOLUTION CHANGES DURING PART 6 =           0.
 THE NUMBER OF TIMES THROUGH PART 7 =                    30.
 THE NUMBER OF SOLUTION CHANGES DURING PART 7 =           0.
AT THE END OF THE HEURISTIC PROCEDURE WE HAVE THE FOLLOWING RESULTS:
 THE FINAL APPROXIMATE INTEGER SOLUTION IS:
   2.00    1.00    0.0     6.00    0.0     1.00    0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     1.00
  121.00
 THE FEASIBILITY TEST SLACKS ARE:
   2.34    0.38    0.25    3.57    0.83    0.37    3.91    0.18    3.94    0.14    0.93    0.00    0.40    0.21    0.97
 THE OBJECTIVE FUNCTION VALUE OF THE FINAL SOLUTION =                                                    7883.00.
 THE NORMALIZED DIFFERENCE BETWEEN THE XONE AND XTWO OBJECTIVE FUNCTION VALUES =                        10.6700.
 THE NORMALIZED DIFFERENCE BETWEEN THE OPTIMAL LP AND INTEGER OBJECTIVE FUNCTION VALUES =                1.2890.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE SINCE THE FIRST TIME THROUGH MODE            0.4004.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE IN PHASE 3 =                                 0.4004.
PHASE 1. METHOD 1.  PHASE 2. METHOD 2.
PHASE 2. CRITERION 4.
THE CURRENT VALUE OF ALPHA IS       0.0 .
THE PHASE 1 SOLUTION XTWO IS:
```

```
   2.54     0.89     0.0      7.29     0.0     -1.27     0.0      0.0      0.0     -2.14     0.0      0.0      0.0      0.0      0.77
  114.78
AT THE END OF PHASE 2 WE HAVE THE FOLLOWING DATA:
 THE VALUE OF ALPHA WHERE THE FEASIBLE SOLUTION WAS FOUND =        0.0   .
 THE NUMBER OF POINTS MOVED TO ON THE SEARCH LINE =                      0.
 THE NUMBER OF TRIAL SOLUTIONS BEFORE FINDING THE FEASIBLE SOLU          4.
 THE SOLUTION FROM PHASE 2 IS:
   3.00     1.00     0.0      6.00     0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0
  121.00
FOR THE FIRST TIME THROUGH MODE 1 WE HAVE THE FOLLOWING DATA:
 THE NUMBER OF SOLUTION CHANGES =                                        0.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE =      0.0   .
THE FOLLOWING OCCURRED PRIOR TO THE TERMINATION OF PHASE 3:
 DURING PHASE 3 WE ENTERED MODE 1    2 TIMES AND HAD    0 SOLUTION CHANGES.
 THE NUMBER OF TIMES THROUGH PART 4 =                                   32.
 THE NUMBER OF SOLUTION CHANGES DURING PART 4 =                          0.
 THE NUMBER OF TIMES THROUGH PART 5 =                                   88.
 THE NUMBER OF SOLUTION CHANGES DURING PART 5 =                          2.
 THE NUMBER OF TIMES THROUGH PART 6 =                                   33.
 THE NUMBER OF SOLUTION CHANGES DURING PART 6 =                          0.
 THE NUMBER OF TIMES THROUGH PART 7 =                                   30.
 THE NUMBER OF SOLUTION CHANGES DURING PART 7 =                          0.
AT THE END OF THE HEURISTIC PROCEDURE WE HAVE THE FOLLOWING RESULTS:
 THE FINAL APPROXIMATE INTEGER SOLUTION IS:
   3.00     1.00     0.0      5.00     0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0
  119.00
 THE FEASIBILITY TEST SLACKS ARE:
   3.46     0.69     0.06     4.05     0.41     0.07     4.46     0.03     4.79     0.07     0.64     0.22     0.48     0.23     0.61
 THE OBJECTIVE FUNCTION VALUE OF THE FINAL SOLUTION =                                                                    7824.00.
 THE NORMALIZED DIFFERENCE BETWEEN THE XONE AND XTWO OBJECTIVE FUNCTION VALUES =                                         4.3132.
 THE NORMALIZED DIFFERENCE BETWEEN THE OPTIMAL LP AND INTEGER OBJECTIVE FUNCTION VALUES =                                1.6265.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE SINCE THE FIRST TIME THROUGH MODE                            0.0343.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE IN PHASE 3 =                                                 0.0343.
PHASE 1. METHOD 2.  PHASE 2. METHOD 2.
PHASE 2. CRITERION 4.
 THE CURRENT VALUE OF ALPHA IS       0.0   .
 THE PHASE 1 SOLUTION XTWO IS:
   2.93     0.72     0.0      5.93     0.0      1.79     0.0      0.0      0.0      1.18     0.0      0.0      0.0      0.0      0.77
  91.67
AT THE END OF PHASE 2 WE HAVE THE FOLLOWING DATA:
 THE VALUE OF ALPHA WHERE THE FEASIBLE SOLUTION WAS FOUND =        0.0   .
 THE NUMBER OF POINTS MOVED TO ON THE SEARCH LINE =                      0.
 THE NUMBER OF TRIAL SOLUTIONS BEFORE FINDING THE FEASIBLE SOLU          4.
 THE SOLUTION FROM PHASE 2 IS:
   3.00     1.00     0.0      6.00     0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0
  121.00
FOR THE FIRST TIME THROUGH MODE 1 WE HAVE THE FOLLOWING DATA:
 THE NUMBER OF SOLUTION CHANGES =                                        0.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE =      0.0   .
THE FOLLOWING OCCURRED PRIOR TO THE TERMINATION OF PHASE 3:
 DURING PHASE 3 WE ENTERED MODE 1    2 TIMES AND HAD    0 SOLUTION CHANGES.
 THE NUMBER OF TIMES THROUGH PART 4 =                                   32.
 THE NUMBER OF SOLUTION CHANGES DURING PART 4 =                          0.
 THE NUMBER OF TIMES THROUGH PART 5 =                                   88.
 THE NUMBER OF SOLUTION CHANGES DURING PART 5 =                          2.
 THE NUMBER OF TIMES THROUGH PART 6 =                                   33.
 THE NUMBER OF SOLUTION CHANGES DURING PART 6 =                          0.
 THE NUMBER OF TIMES THROUGH PART 7 =                                   30.
 THE NUMBER OF SOLUTION CHANGES DURING PART 7 =                          0.
AT THE END OF THE HEURISTIC PROCEDURE WE HAVE THE FOLLOWING RESULTS:

 THE FINAL APPROXIMATE INTEGER SOLUTION IS:
   3.00     1.00     0.0      5.00     0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0      0.0
  119.00
 THE FEASIBILITY TEST SLACKS ARE:
   3.46     0.69     0.06     4.05     0.41     0.07     4.46     0.03     4.79     0.07     0.64     0.22     0.48     0.23     0.61
 THE OBJECTIVE FUNCTION VALUE OF THE FINAL SOLUTION =                                                                    7824.00.
 THE NORMALIZED DIFFERENCE BETWEEN THE XONE AND XTWO OBJECTIVE FUNCTION VALUES =                                        10.6700.
 THE NORMALIZED DIFFERENCE BETWEEN THE OPTIMAL LP AND INTEGER OBJECTIVE FUNCTION VALUES =                                1.6265.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE SINCE THE FIRST TIME THROUGH MODE                            0.0343.
 THE NORMALIZED IMPROVEMENT IN THE OBJECTIVE FUNCTION VALUE IN PHASE 3 =                                                 0.0343.
```

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER No. 88	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) DOCUMENTATION OF A COMPUTER PROGRAM FOR HILLIER'S HEURISTIC PROCEDURE IN INTEGER LINEAR PROGRAMMING		5. TYPE OF REPORT & PERIOD COVERED TECHNICAL REPORT 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) NANCY E. JACOMIN		8. CONTRACT OR GRANT NUMBER(s) N00014-76-C-0418
9. PERFORMING ORGANIZATION NAME AND ADDRESS DEPT. OF OPERATIONS RESEARCH STANFORD UNIVERSITY, STANFORD, CA.		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NR-047-061
11. CONTROLLING OFFICE NAME AND ADDRESS OPERATIONS RESEARCH PROGRAM OFFICE OF NAVAL RESEARCH CODE 434 ARLINGTON, VA. 22217		12. REPORT DATE JULY 1979 13. NUMBER OF PAGES 55
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

This research was supported in part by National Science Foundation Grant NSF Number - MCS 76-81259 A01, Dept. of Operations Research Stanford University.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

INTEGER PROGRAMMING, HEURISTICS, SOFTWARE

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report gives documentation for the computer code HEUR for finding good approximate solutions to integer linear programming problems using Hillier's heuristic procedure. A listing of the program with sample input and output is included.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)