NAME

SIMPLX − Use the simplex algorithm to solve a linear program in standard form.

SYNOPSIS

CALL SIMPLX(T,LDT,MP,N,ROWS,MR, BASIS,X,RC)

T(LDT,*)	is the REAL*8 input LP tableau
LDT	is the INTEGER*4 leading dimension of T
MP	is the INTEGER*4 total number of rows in the LP including the objective
N	is the INTEGER*4 number of variables in the LP
ROWS(MR)	is the INTEGER*4 input vector of the row indices in the problem
MR	is the INTEGER*4 input number of rows in use including the objective
BASIS(N)	is the INTEGER*4 output vector of basic column row indices, 0 for nonbasic
X(N)	is the REAL*8 solution vector
RC	is the INTEGER*4 return code; see below

DESCRIPTION

The routine returns RC=−1 without doing anything if the input parameters do not make sense. Otherwise it solves the linear program using the approach described in [1], which involves pivoting in the tableau T. First it calls PHASE0 to get the identity columns with zero costs above them; if this process fails it returns RC=1. Then it uses PHASE1 to get the constant column nonnegative by the subproblem technique; if this process fails it returns RC=1. Next it uses PHASE2 to get optimal form. Finally it extracts the solution vector from the optimal form tableau and returns, with RC set to the return code from PHASE2.

DIAGNOSTICS

On output these are the possible RC values:

−1	the input parameters do not make sense
0	all went well
1	the problem is infeasible
>1	the problem is unbounded in column RC

LINKAGE

gfortran source.f −L${HOME}/lib −lmisc

AUTHOR

Michael Kupferschmid

REFERENCE

[1] Kupferschmid, Michael, "Introduction to Mathematical Programming"

EXAMPLE

```
      PARAMETER(LDT=4,MP=4,N=7,MR=4)
      INTEGER*4 ROWS(MR)/1,2,3,4/,BASIS(7),RC
      REAL*8 T(LDT,1+N)/0.0D0, 1.6D2, 5.0D1, 6.0D1,
     ;                 -9.0D1, 7.0D0, 1.0D0, 2.0D0,
     ;                 -1.5D2, 1.0D1, 3.0D0, 4.0D0,
     ;                 -6.0D1, 8.0D0, 1.0D0, 1.0D0,
     ;                 -7.0D1, 1.2D1, 1.0D0, 3.0D0,
     ;                  0.0D0, 1.0D0, 0.0D0, 0.0D0,
     ;                  0.0D0, 0.0D0, 1.0D0, 0.0D0,
     ;                  0.0D0, 0.0D0, 0.0D0, 1.0D0/
      REAL*8 X(N)
      DO 1 I=1,MP
           WRITE(6,901) (T(I,J),J=1,1+N)
  901      FORMAT(8(1X,F7.1))
    1 CONTINUE
      WRITE(6,900)
  900 FORMAT(' ')
      CALL SIMPLX(T,LDT,MP,N,ROWS,MR, BASIS,X,RC)
      DO 2 I=1,MP
           WRITE(6,901) (T(I,J),J=1,1+N)
    2 CONTINUE
      WRITE(6,900)
      WRITE(6,902) (X(J),J=1,N)
  902 FORMAT(8X,7(1X,F7.1))
      STOP
      END
```

This example (which is from [1] page 72) produced the following output:

```
unix[1] a.out
     0.0   -90.0  -150.0   -60.0   -70.0     0.0     0.0     0.0
   160.0     7.0    10.0     8.0    12.0     1.0     0.0     0.0
    50.0     1.0     3.0     1.0     1.0     0.0     1.0     0.0
    60.0     2.0     4.0     1.0     3.0     0.0     0.0     1.0

  2325.0     0.0     0.0    18.8    76.3     7.5     0.0    18.8
     5.0     1.0     0.0     2.8     2.3     0.5     0.0    -1.2
    12.5     0.0     1.0    -1.1    -0.4    -0.3     0.0     0.9
     7.5     0.0     0.0     1.6    -0.1     0.3     1.0    -1.4

             5.0    12.5     0.0     0.0     0.0     7.5     0.0
unix[2]
```