

ADAS Subroutine lumsis

SUBROUTINE LUMSIS(N)

IMPLICIT REAL*8 (A-H,O-Z)

```
C-----
C PURPOSE: Finds the solutions of a set of linear equations
C
C N- a number of equations in the set + 1
C AK(I,J)-coefficients in equations
C BK(I) - right side terms of equations
C When solutions are found they will be written into array BK(I) in
C the common block
C
C-----
C ADAS305 version. Developed from JETSHP.STARK.FORT (H P Summers).
C
C VERSION   : 1.1
C DATE     : 24-02-2005
C MODIFIED : Martin O'Mullane
C           - First version.
C
C VERSION   : 1.2
C DATE     : 16-05-2007
C MODIFIED : Allan Whiteford
C           - Updated comments as part of subroutine documentation
C           procedure.
C
C-----
COMMON /CX/ AK(65,65),BK(65)
C*****
C**   EXPANSION OF MATRIX AK
C*****

      DO 200 I=1,N-1
      AK(N,I)=FLOAT(I)
200   AK(I,N)=BK(I)

C*****   MAX OF MAIN MINORS

      DO 202 K=1,N-2
      AMAX=AK(K,K)
      IMAX=K
      JMAX=K
      DO 206 I=K,N-1
      DO 206 J=K,N-1
      IF (DABS(AMAX).GE.DABS(AK(I,J))) GOTO 206
      AMAX=AK(I,J)
      JMAX=J
      IMAX=I
206   CONTINUE

C*****   LINES INTERCHANGE
```

```

        IF (IMAX.EQ.K) GOTO 210
        DO 212 J1=1,N
        AB=AK (IMAX,J1)
        AK (IMAX,J1)=AK (K,J1)
212    AK (K,J1)=AB

C***** ROWS INTERCHANGE

210    IF (JMAX.EQ.K) GOTO 202
        DO 216 I1=1,N
        AB=AK (I1,JMAX)
        AK (I1,JMAX)=AK (I1,K)
216    AK (I1,K)=AB
202    CONTINUE

C***** FIRST FORMATION BK

        DO 218 I=1,N-1
218    BK (I)=AK (I,N)

C*****
C***** LU EXPANSION
C*****

        DO 120 K=1,N-2
        AB=1.E00/AK (K,K)
        DO 122 J=K,N-2
122    AK (K,J+1)=AK (K,J+1)*AB

        DO 120 J=K+1,N-1
        DO 120 I=K+1,N-1
120    AK (I,J)=AK (I,J)-AK (I,K)*AK (K,J)

C***** SOLUTION OF THE SYSTEM

        BK (1)=BK (1)/AK (1,1)
        DO 105 I=2,N-1
        DO 106 J=1,I-1
106    BK (I)=BK (I)-AK (I,J)*BK (J)
105    BK (I)=BK (I)/AK (I,I)

        DO 107 I=N-2,1,-1
        DO 107 J=I+1,N-1
107    BK (I)=BK (I)-AK (I,J)*BK (J)

C***** SECOND FORMATION OF BK

        DO 240 J=1,N-1
        DO 241 IB=1,N-1
        IF (IB.NE.IDINT (DABS (AK (N,J)))) GOTO 241
        II=IB
        GOTO 240

```

```
241  CONTINUE
240  AK (II, N) =BK (J)
      DO 242 J=1, N-1
242  BK (J) =AK (J, N)
      RETURN
      END
      INTEGER          N
```