1 SUMMARY

This subroutine obtains a bound on the largest entry encountered during Gaussian elimination on a sparse matrix $\mathbf{A} = \{a_{ij}\}_{n \times n}$. The subroutine is given the **LU** decomposition factors of **A** obtained from the elimination. If the matrix has been scaled, this estimate will give an indication of the numerical accuracy of the decomposition.

The estimate is obtained by using Hadamard's inequality on the expression for the matrix elements in terms of the **LU** factors of the decomposition. Further details are given in Erisman and Reid, 'Monitoring the Stability of the Triangular Factorization of a Sparse Matrix', Numer. Math. **22** (1974).

The code is described by I.S.Duff, Harwell report R.8730 (1977).

ATTRIBUTES — **Version:** 1.0.0. **Types:** `MC24A`, `MC24AD`. **Original date:** March 1977. **Origin:** I. S. Duff, Harwell.

2 HOW TO USE THE PACKAGE

2.1 Argument list

The single precision version

```
CALL MC24A(N,ICN,A,LICN,LENR,LENRL,W)
```

The double precision version

```
CALL MC24AD(N,ICN,A,LICN,LENR,LENRL,W)
```

N   is an `INTEGER` variable which must be set by the user to the order *n* of the matrix **A**. It is not altered by the subroutine. **Restriction:** $n \geq 1$.

ICN   is an `INTEGER` array of length `LICN` which must contain the column indices of the nonzeros in the decomposition. Each row (of **L** and **U**) is held contiguously. Row `I` precedes row `I+1`, `I=1,...,N-1` and there is no wasted space between the rows. Although the column indices need not be in order, those in **L** must precede those in **U** with the pivot being the first entry in the row of **U**. It is not altered by the subroutine.

A   is a `REAL` (`DOUBLE PRECISION` in the D version) array of length `LICN` which contains the values of the entries in the **LU** decomposition. The nonzero held in `A(K)` is in column `ICN(K)`. It is not altered by the subroutine.

LICN   is an `INTEGER` variable which must be set by the user to be the length of arrays `ICN` and `A`. It is not altered by the subroutine.

LENR   is an `INTEGER` array of length `N`. `LENR(I)` must be set by the user to the combined number of nonzeros in rows `I` of **L** and **U**, `I=1,...,N`. It is not altered by the subroutine.

LENRL   is an `INTEGER` array of length `N`. `LENRL(I)` must be set by the user to the number of nonzeros in row `I` of **L**, `I=1,2,...,N`. It is not altered by the subroutine.

W   is a `REAL` (`DOUBLE PRECISION` in the D version) array of length `N`. It is used as workspace and, on output, `W(1)` is equal to the estimate of the largest entry encountered during the **LU** decomposition.
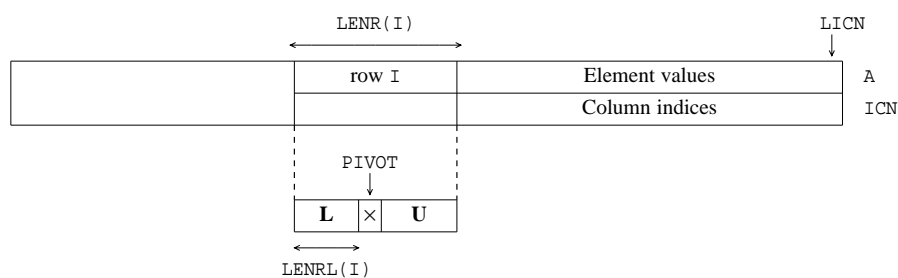
2.2 Parameter usage summary

Input: N, ICN(LICN), A(LICN), LICN, LENR(N), LENRL(N)

Unchanged by MC24A: N, ICN, A, LICN, LENR, LENRL

Work array: W(N)

Output: W(1)

2.3 Data structure summary

3 GENERAL INFORMATION

Use of common: None.

Workspace: W of length N.

Other routines called directly: None.

Input/output: None.

Restrictions: $n \geq 1$.

4 METHOD

The estimate is obtained by using Hadamard's inequality on the expression for the matrix entries in terms of the **LU** factors of the decomposition. For further details, the reader is referred to Erisman and Reid, Monitoring the stability of the triangular factorization of a sparse matrix, Numer. Math. **22** (1974), 183-186.

5 EXAMPLE OF USE

This example is particularly simple. It is envisaged that in practice the subroutine will be called after code performing the **LU** decomposition. An option in MA28A/AD allows MC24A/AD to be called by it.

```
      REAL  A(29),W(3)
      INTEGER ICN(20),LENR(3),LENRL(3)
      N=3
      NZ=7
      LICN=20
C READ IN INPUT MATRIX
      READ(5, * ) (LENR(I),LENRL(I),I=1,N)
      READ(5, * ) (A(I), ICN(I), I=1,NZ)
C PERFORM GROWTH ESTIMATE
      CALL MC24A(N,ICN,A,LICN,LENR,LENRL,W)
C PRINT OUT ESTIMATE OF INCREASE IN ELEMENT SIZE
      WRITE(6,30) W(1)
 30   FORMAT(' ESTIMATE OF LARGEST ELEMENT ENCOUNTERED '/
     +       '             DURING GAUSSIAN ELIMINATION IS ' , E15.6 )
      STOP
      END
```

If the data input to the above program were

```
    2   0
    3   1
    2   1
  0.510000E+00   1
  0.230000E+01   2
 -0.312000E+01   1
  0.320000E+00   2
  0.450000E+01   3
  0.130000E+01   2
 -0.240000E+01   3
```

corresponding to the matrices:

$$\mathbf{L} = \begin{pmatrix} 1.00 & & \\ -3.12 & 1.00 & \\ 0.00 & 1.30 & 1.00 \end{pmatrix} \quad \text{and} \quad \mathbf{U} = \begin{pmatrix} 0.51 & 2.30 & 0.00 \\ & 0.32 & 4.50 \\ & & -2.40 \end{pmatrix}$$

then the output would be

```
  ESTIMATE OF LARGEST ELEMENT ENCOUNTERED
          DURING GAUSSIAN ELIMINATION IS   0.140400E+02
```