Warning: Subroutine MC15 performs functions which are adequately treated by routines in other standard subroutine libraries (for example, LAPACK). The use of this routine is not recommended, and it may be removed from future releases of this library.

1 SUMMARY

To **scale** the rows and columns of a given n by n **real matrix A** so that its elements are suitably balanced for accurate computation of its eigenvalues and eigenvectors.

Diagonal similarity transformations are used based on powers of 16 so that on a hexadecimal based computer only the exponent fields of the matrix elements are modified. Searches along rows and columns for isolated eigenvalues are also made and the matrix permuted so as to group such eigenvalues together. For the method see J.H. Wilkinson and C. Reinsch, 'Handbook for Automatic Computation – Linear Algebra', Springer-Verlag.

ATTRIBUTES — **Version:** 1.0.0. **Types:** MC15A; MC15AD. **Original date:** June 1973. **Origin:** BALANCE from the above reference, modified for Harwell by S.Marlow.

2 HOW TO USE THE PACKAGE

2.1 The argument list

The single precision version

```
CALL MC15A(A,IA,N,W,LL,LU)
```

The double precision version

```
CALL MC15AD(A,IA,N,W,LL,LU)
```

A is a REAL (DOUBLE PRECISION in the D version) two dimensional array which must be set by the user to contain the elements of the matrix. It will be overwritten by the balanced matrix, see LL and LU below.

IA is an INTEGER variable which must be set by the user to the first dimension of the array A. **Restriction:** IA $\geq n$.

N is an INTEGER variable which must be set by the user to n the order of the matrix.

W is a REAL (DOUBLE PRECISION in the D version) array of length n which will be set by the subroutine to contain information determining the permutations used and the scaling factors. See Section 4.

LL and LU are both INTEGER variables which are set by the subroutine. Their values are such that A(i,j) is zero if i > j and j=1, 2, ..., LL-1 or i=LU+1, ..., N, i.e. the balanced matrix will be of the form

$$\begin{bmatrix} \mathbf{U}_{1,1} & \mathbf{F}_{1,2} & \mathbf{F}_{1,3} \\ 0 & \mathbf{F}_{2,2} & \mathbf{F}_{2,3} \\ 0 & 0 & \mathbf{U}_{3,3} \end{bmatrix}$$

where $\mathbf{U}_{1,1}$ and $\mathbf{U}_{3,3}$ are upper triangular matrices of orders LL-1 and N-LU, and $\mathbf{F}_{1,2}$, $\mathbf{F}_{1,3}$, $\mathbf{F}_{2,2}$ and $\mathbf{F}_{2,3}$ are in general full matrices with $\mathbf{F}_{2,2}$ of order LU+LL+1.

2.2 The common areas

The subroutine uses a labelled common block called MC15B/BD which is initialized through a BLOCK DATA subroutine. If the user wishes to change the common block values then the COMMON statement which follows should be

included and the appropriate values reset.

The single precision version

```
COMMON/MC15B/ LP
```

The double precision version

```
COMMON/MC15BD/ LP
```

`LP` is an `INTEGER` variable (default 6) and refers to the stream number on which diagnostic messages appear. To suppress these messages set `LP` to 0.

3 GENERAL INFORMATION

Use of common: uses `MC15B/BD` (see §2.2).

Workspace: None.

Other routines called directly: None.

Input/output: diagnostic printing (see §2.2).

4 METHOD

First an attempt is made to group together any isolated eigenvalues in the matrix **A**. To do this a search is made for a row of **A** which contains only one non-zero element which is on the diagonal of **A**. If there is such a row, permutations are applied to **A**, i.e. $\mathbf{PAP}^T$ to move the non-zero element to the bottom right hand corner. The search is then continued looking for a row in the remaining $(n-1)\times(n-1)$ matrix which has only one non-zero element on the diagonal. If there is one it is in turn permuted to the $(n-1,n-1)$ position in **A**. This process is continued on each reduced matrix until no more rows are found with only one non-zero element on the diagonal. At this point the permuted form of **A** will have at the bottom right hand corner an upper triangle submatrix whose diagonals will be isolated eigenvalues. Next the same procedure will be applied to the columns of **A**, looking for columns with only one non-zero element on the diagonal of **A** and these isolated eigenvalues will be permuted into an upper triangle submatrix in the upper left hand corner of **A**. Each permutation consists of interchanging two columns i and p_i say, and the corresponding two rows. The value of p_i will be found in `W(i)`. The interchanges are performed in the order i = `N` to `LU+1` and i = 1 to `LL-1`. Balancing of the matrix is performed by reducing the norm of the matrix by exact diagonal similarity transformations to $\mathbf{D}^{-1}\mathbf{P}^T\mathbf{APD}$ where **D** is a diagonal matrix with elements (1, 1, 1,... 1, `W(LL),...W(LU),` 1, 1, 1).