> **Warning:** Subroutine `EB06` 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

Finds **all the eigenvalues and right eigenvectors of a general** n **by** n **real matrix,** i.e. given $\mathbf{A} = \{a_{ij}\}_{n \times n}$, finds solutions λ_i and $\mathbf{z}_i$ of

$$\mathbf{A}\mathbf{z}_i = \lambda_i \mathbf{z}_i, \qquad i=1,2,\dots,n.$$

There is an option for scaling the rows and columns of $\mathbf{A}$ so that its elements are suitably balanced for accurate computation of its eigenvalues and eigenvectors. The vectors are normalised to have unit length, i.e.

$$\sum_{i=1}^{n} z_i \bar{z}_i = 1.$$

A QR method is used. The matrix is reduced to Hessenberg form by orthogonal similarity transformations (`MC14A/AD`) and the QR method is applied to obtain the eigenvalues (`EB08A/AD`), see J.H.Wilkinson and C.Reinsch, 'Handbook for Automatic Computation, Linear Algebra', Springer-Verlag. Balancing is performed by `MC15A`.

ATTRIBUTES — **Version:** 1.0.0. **Types:** `EB06A`; `EB06AD`. **Calls:** `EB08`, `MC14` and `MC15`. **Original date:** June 1973. **Origin:** S.Marlow, Harwell.

2 HOW TO USE THE PACKAGE

2.1 The argument list and calling sequence

The single precision version:

```
CALL EB06A(A,IA,Z,IZ,E,N,W,LFAIL)
```

The double precision version:

```
CALL EB06AD(A,IA,Z,IZ,E,N,W,LFAIL)
```

A   is a two-dimensional REAL (DOUBLE PRECISION in the D version) array of dimensions at least n x n (first dimension specified in IA), which must be set by the user to the elements of the matrix. It is overwritten by the subroutine.

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

Z   is a two-dimensional REAL (DOUBLE PRECISION in the D version) array of dimensions at least n x n (first dimension specified in IZ), which will be set by the subroutine to the eigenvectors. If the ith eigenvalue is real then the ith column of Z will contain the corresponding eigenvector. If the ith and $(i+1)^{\text{th}}$ eigenvalues in E are a complex conjugate pair the columns i and $i+1$ of Z give the real and imaginary part of the eigenvector corresponding to the eigenvalue with positive imaginary part. (The imaginary part of E(i) will be zero if the ith eigenvalue is real).

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

E   is a two-dimensional REAL (DOUBLE PRECISION in the D version) array of first dimension 2 which will be set

by the subroutine to the eigenvalues. `E(1,I)`, `E(2,I)` containing the real and imaginary parts of the `I` th eigenvalue. Complex conjugate pairs will be stored adjacent to each other.

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

W   is a one-dimensional `REAL` (`DOUBLE PRECISION` in the D version) array of length at least 2n which will be used by the subroutine as working space.

LFAIL is an `INTEGER` variable which is an error return parameter and will be set by the subroutine to have one of the following values

   0 successful return
   –1 value of `N` not positive
  $k > 0$ the QR method failed to converge for the kth eigenvalue

2.2 The Common area

The subroutine uses a Common area which the user may reference by inserting a Common statement of the form

The single precision version:

```
COMMON/EB06B/ LP,LB,ITS
```

The double precision version:

```
COMMON/EB06BD/ LP,LB,ITS
```

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

LB   is an `INTEGER` variable (default 0) that indicates that no matrix balancing is done. If balancing is required its value must be nonzero.

ITS   is an `INTEGER` variable (default 30) and is the limit on the number of iterations allowed per eigenvalue.

3 GENERAL INFORMATION

Use of Common: uses common areas `EB06B/BD`, see §2.2.

Workspace: 2n words provided in `W`.

Other subroutines: calls `EB08A/AD, _DOT, MC14A/AD` and `MC15A/AD`.

Input/Output: error messages will be printed, see §2.2, when either N≤0 or there is a failure of convergence.

4 METHOD

The matrix is first balanced if requested by the user (`MC15A/AD`). The matrix is then reduced to upper Hessenberg form using orthogonal similarity transformations (`MC14A/AD`). The eigenvalues of the Hessenberg form are found by applying the QR, at the same time the eigenvectors are formed by accumulating the transformations (`EB08A/AD`). The eigenvectors of the full matrix are then found by back-transformation. If balancing was requested, then the eigenvectors of the balanced matrix are then back-transformed into the eigenvectors of the original matrix.

5 EXAMPLE OF USE

The eigenvectors, which in general will be complex, are returned by `EB06A` in a compact `REAL` form. The example shown below has been included to illustrate how a user might extract them as complex vectors.

In the example we have called `EB06A` to obtain the eigenvectors of a 10 by 10 matrix **A**, to unpack the eigenvectors from the real array `Z` and then store them as complex vectors in the complex array `C`.

```
      REAL Z(20,20), A(14,14), E(2,20), W(28)
      COMPLEX C(20,20)
         .
         .
         .
         .
      N=10
      CALL EB06A(A,14,Z,20,E,N,W,LFAIL)
      IF(LFAIL.NE.0) GO TO 50
      J=0
      DO 40 I=1,N
C     LOOK AT ITH EIGENVALUE AND SEE WHETHER IT IS REAL OR COMPLEX
      EI=IMAG(E(I))
C     IGNORE COMPLEX EIGENVALUE WITH NEGATIVE IMAGINARY PART
      IF(EI) 40,20,10
C     EIGENVALUE IS COMPLEX WITH POSITIVE IMAGINARY PART
   10 J=J+2
      DO 15 K=1,N
      C(K,J-1)=CMPLX(Z(K,J-1),Z(K,J))
   15 C(K,J)=CONJG(C(K,J-1))
      GO TO 40
C     REAL EIGENVECTOR
   20 J=J+1
      DO 30 K=1,N
   30 C(K,J)=Z(K,J)
   40 CONTINUE
         .
         .
         .
   50 -
C     DEAL WITH ERRORS
```