1 SUMMARY

Given a **real symmetric matrix A** of order *n*, finds **all its eigenvalues** λ_i **and eigenvectors** $\mathbf{x}_i$, *i*=1,2,...,*n*, where $\mathbf{A}\mathbf{x}_i = \lambda_i \mathbf{x}_i$.

The classical Jacobi method is used.

ATTRIBUTES — **Version:** 1.1.0. **Types:** Real (single, double). **Original date:** May 2001. **Remark:** EA23 is a threadsafe version of EA13. **Origin:** I.S.Duff, Harwell.

2 HOW TO USE THE PACKAGE

2.1 Argument list

The single precision version

```
CALL EA23A(N,A,ND,D,V,EIGVEC,IROT,IW,LP)
```

The double precision version

```
CALL EA23AD(N,A,ND,D,V,EIGVEC,IROT,IW,LP)
```

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

A is a REAL (DOUBLE PRECISION in the D version) two-dimensional array of dimensions (ND,N). On entry, the upper triangular part of the leading principal submatrix of order N must contain the matrix whose eigensystem the user wishes to calculate. It will have its entries altered by EA23A/AD.

ND is an INTEGER variable which must be set by the user to the first dimension of arrays A and V.

D is a REAL (DOUBLE PRECISION in the D version) array of length N. On output from EA23A/AD, it will hold the eigenvalues of the matrix **A**.

V is a REAL (DOUBLE PRECISION in the D version) two-dimensional array of dimensions (ND,N). On output from EA23A/AD, it will hold the eigenvectors of the matrix **A**, its I-th column holding the eigenvector corresponding to eigenvalue of **A** in D(I), I=1,...,N.

EIGVEC is a LOGICAL variable. It should be set to true by the user if he or she wishes the eigenvectors as well as the eigenvalues of **A** and to false if he or she only wants the eigenvalues.

IROT is an INTEGER variable whose value should be set by the user to the maximum number of Jacobi rotations he or she wishes to allow. If this number is exceeded, an error message is printed (see section 2.2); on successful completion its output value equals the number of rotations actually performed. For guidance on how to choose IROT see section 4.

IW is an INTEGER work array of length N.

LP is an INTEGER variable to be used by the routine when writing error messages. It must be set to a valid unit number (normally 6), or set to zero to suppress messages.

2.2 Error return

There is only one error return which occurs if the algorithm has not converged after the IROT rotations specified by the user. If this happens the message

```
TOO MANY ITERATIONS TAKEN BY EA23A/AD
```

will be output on the unit given by the argument LP and IROT will be output with a value equal to its input value.

3 GENERAL INFORMATION

Use of common: None.

Workspace: Argument IW of length N.

Other routines called directly: None.

Input/output: error messages are printed on the unit given by the argument LP, see section 2.2.

4 METHOD

The method used is a classical version of Jacobi's method. At each stage the largest off-diagonal element, (p,q) say, is reduced to zero by a rotation in the (p,q)-th plane. The process stops when all off-diagonal elements are small compared with the diagonal elements.

The method is quadratically convergent and will, in normal cases, require only a small multiple of n^2 rotations for convergence.

5 EXAMPLE OF USE

Suppose that we are given the symmetric matrix

$$\begin{bmatrix} 1.2 & 0.5 & 1.3 & 0.3 \\ 0.5 & 0.9 & 0.0 & 2.1 \\ 1.3 & 0.0 & 1.7 & 1.7 \\ 0.3 & 2.1 & 1.7 & 0.5 \end{bmatrix}$$

and we wish to compute all the eigenvalues. The Fortran code for doing this might be

```
DOUBLE PRECISION D(4),A(4,4),V(4,4)
INTEGER IROT,IW(4),J,N,ND
LOGICAL EIGVEC
DATA A(1,1),A(1,2),A(1,3),A(1,4)/1.2D0,0.5D0,1.3D0,0.3D0/
DATA A(2,2),A(2,3),A(2,4)       /      0.9D0,0.0D0,2.1D0/
DATA A(3,3),A(3,4)              /            1.7D0,1.7D0/
DATA A(4,4)                     /                  0.5D0/
N=4
ND=4
EIGVEC=.TRUE.
IROT=50
CALL EA23AD (N,A,ND,D,V,EIGVEC,IROT,IW,6)
WRITE (6,'(A,1P,4E12.4)') ' Eigenvalues are',(D(J),J=1,N)
END
```

The output from the code is

```
 Eigenvalues are  4.7504E-01  4.1350E+00  1.6722E+00 -1.9823E+00
```