NAG Toolbox

nag_lapack_zheev (f08fn)

1 Purpose

nag_lapack_zheev (f08fn) computes all the eigenvalues and, optionally, all the eigenvectors of a complex n by n Hermitian matrix A.

2 Syntax

```
[a, w, info] = nag_lapack_zheev(jobz, uplo, a, 'n', n)
[a, w, info] = f08fn(jobz, uplo, a, 'n', n)
```

3 Description

The Hermitian matrix A is first reduced to real tridiagonal form, using unitary similarity transformations, and then the QR algorithm is applied to the tridiagonal matrix to compute the eigenvalues and (optionally) the eigenvectors.

4 References

Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J J, Du Croz J J, Greenbaum A, Hammarling S, McKenney A and Sorensen D (1999) *LAPACK Users' Guide* (3rd Edition) SIAM, Philadelphia http://www.netlib.org/lapack/lug

Golub G H and Van Loan C F (1996) *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

5 Parameters

5.1 Compulsory Input Parameters

1: **jobz** – CHARACTER(1)

Indicates whether eigenvectors are computed.

jobz = 'N'
Only eigenvalues are computed.

jobz = 'V'
Eigenvalues and eigenvectors are computed.

Constraint: **jobz** = 'N' or 'V'.

2: **uplo** – CHARACTER(1)

If **uplo** = 'U', the upper triangular part of A is stored.

If **uplo** = 'L', the lower triangular part of A is stored.

Constraint: **uplo** = 'U' or 'L'.

3: **a**$(lda, :)$ – COMPLEX (KIND=nag_wp) array

The first dimension of the array **a** must be at least $\max(1, \mathbf{n})$.

The second dimension of the array **a** must be at least $\max(1, \mathbf{n})$.

The n by n Hermitian matrix A.

If **uplo** = 'U', the upper triangular part of a must be stored and the elements of the array below the diagonal are not referenced.

If **uplo** = 'L', the lower triangular part of a must be stored and the elements of the array above the diagonal are not referenced.

5.2 Optional Input Parameters

1: **n** – INTEGER

Default: the first dimension of the array **a** and the second dimension of the array **a**. (An error is raised if these dimensions are not equal.)

n, the order of the matrix A.

Constraint: **n** ≥ 0.

5.3 Output Parameters

1: **a**$(lda,:)$ – COMPLEX (KIND=nag_wp) array

The first dimension of the array **a** will be $\max(1,\mathbf{n})$.

The second dimension of the array **a** will be $\max(1,\mathbf{n})$.

If **jobz** = 'V', then **a** contains the orthonormal eigenvectors of the matrix A.

If **jobz** = 'N', then on exit the lower triangle (if **uplo** = 'L') or the upper triangle (if **uplo** = 'U') of **a**, including the diagonal, is overwritten.

2: **w**$(\mathbf{n})$ – REAL (KIND=nag_wp) array

The eigenvalues in ascending order.

3: **info** – INTEGER

info = 0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

info $= -i$

If **info** $= -i$, parameter i had an illegal value on entry. The parameters are numbered as follows:

1: **jobz**, 2: **uplo**, 3: **n**, 4: **a**, 5: **lda**, 6: **w**, 7: **work**, 8: **lwork**, 9: **rwork**, 10: **info**.

It is possible that **info** refers to a parameter that is omitted from the MATLAB interface. This usually indicates that an error in one of the other input parameters has caused an incorrect value to be inferred.

info > 0 (*warning*)

If **info** $= i$, the algorithm failed to converge; i off-diagonal elements of an intermediate tridiagonal form did not converge to zero.

7 Accuracy

The computed eigenvalues and eigenvectors are exact for a nearby matrix $(A+E)$, where

$$\|E\|_2 = O(\epsilon)\|A\|_2,$$

and ϵ is the ***machine precision***. See Section 4.7 of Anderson *et al.* (1999) for further details.

8 Further Comments

Each eigenvector is normalized so that the element of largest absolute value is real.

The total number of floating-point operations is proportional to n^3.

The real analogue of this function is nag_lapack_dsyev (f08fa).

9 Example

This example finds all the eigenvalues and eigenvectors of the Hermitian matrix

$$A = \begin{pmatrix} 1 & 2-i & 3-i & 4-i \\ 2+i & 2 & 3-2i & 4-2i \\ 3+i & 3+2i & 3 & 4-3i \\ 4+i & 4+2i & 4+3i & 4 \end{pmatrix},$$

together with approximate error bounds for the computed eigenvalues and eigenvectors.

9.1 Program Text

```
     function f08fn_example

fprintf('f08fn example results\n\n');

% upper triangular part of Hernitian matrix A
n = nag_int(4);
uplo = 'Upper';
a = [ 1,       2 - 1i,  3 - 1i,  4 - 1i;
      0 + 0i,  2 + 0i,  3 - 2i,  4 - 2i;
      0 + 0i,  0 + 0i,  3 + 0i,  4 - 3i;
      0 + 0i,  0 + 0i,  0 + 0i,  4 + 0i];

jobz = 'Vectors';
[z, w, info] = f08fn( ...
       jobz, uplo, a);

disp('Eigenvalues');
disp(w);

% Normalize
for i = 1:n
  [~,k] = max(abs(real(z(:,i)))+abs(imag(z(:,i))));
  z(1:n,i) = z(1:n,i)*conj(z(k,i))/abs(z(k,i));
end
[ifail] = x04da( ...
                 'General', ' ', z, 'Eigenvectors');

% Eigenvalue error bound
errbnd = x02aj*max(abs(w(1)),abs(w(end)));
% Eigenvector condition numbers
[rcondz, info] = f08fl( ...
         'Eigenvectors', n, n, w);

% Eigenvector error bounds
zerrbd = errbnd./rcondz;

fprintf('\n');
disp('Error estimate for the eigenvalues');
fprintf('%12.1e\n',errbnd);
disp('Error estimates for the eigenvectors');
fprintf('%12.1e',zerrbd);
fprintf('\n');
```

9.2 Program Results

```
     f08fn example results

Eigenvalues
   -4.2443
   -0.6886
    1.1412
   13.7916

 Eigenvectors
          1       2       3       4
 1   0.4836  0.6470  0.0179  0.3809
     0.0000  0.0000 -0.4453 -0.0622

 2   0.2912 -0.4984  0.5706  0.4358
    -0.3618 -0.1130  0.0000 -0.0869

 3  -0.3163  0.2949 -0.1530  0.5241
    -0.3696  0.3165  0.5273  0.0000

 4  -0.4447 -0.2241 -0.2118  0.5719
     0.3406 -0.2878 -0.3598  0.2276

Error estimate for the eigenvalues
     1.5e-15
Error estimates for the eigenvectors
     4.3e-16     8.4e-16     8.4e-16     1.2e-16
```