NAME

SSYEVD - computes all eigenvalues and, optionally, eigenvectors of a real symmetric matrix A

SYNOPSIS

SUBROUTINE SSYEVD(
JOBZ, UPLO, N, A, LDA, W, WORK, LWORK, IWORK, LIWORK, INFO)

CHARACTER
JOBZ, UPLO

INTEGER INFO, LDA, LIWORK, LWORK, N

INTEGER IWORK(*)

REAL A(LDA, *), W(*), WORK(*)

PURPOSE

SSYEVD computes all eigenvalues and, optionally, eigenvectors of a real symmetric matrix A. If eigenvectors are desired, it uses a divide and conquer algorithm.

The divide and conquer algorithm makes very mild assumptions about floating point arithmetic. It will work on machines with a guard digit in add/subtract, or on those binary machines without guard digits which subtract like the Cray X-MP, Cray Y-MP, Cray C-90, or Cray-2. It could conceivably fail on hexadecimal or decimal machines without guard digits, but we know of none.

Because of large use of BLAS of level 3, SSYEVD needs N**2 more workspace than SSYEVX.

ARGUMENTS

JOBZ (input) CHARACTER*1
= 'N': Compute eigenvalues only;
= 'V': Compute eigenvalues and eigenvectors.

UPLO (input) CHARACTER*1
= 'U': Upper triangle of A is stored;
= 'L': Lower triangle of A is stored.

N (input) INTEGER
The order of the matrix A. N >= 0.

A (input/output) REAL array, dimension (LDA, N)
On entry, the symmetric matrix A. If UPLO = 'U', the leading N-by-N upper triangular part of A contains the upper triangular part of the matrix A. If UPLO = 'L', the leading N-by-N lower triangular part of A contains the lower triangular part of the matrix A. On exit, if JOBZ = 'V', then if INFO = 0, 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 destroyed.

LDA (input) INTEGER
The leading dimension of the array A. LDA >= max(1,N).

W (output) REAL array, dimension (N)
If INFO = 0, the eigenvalues in ascending order.

WORK (workspace/output) REAL array,
dimension (LWORK) On exit, if INFO = 0, WORK(1) returns the optimal LWORK.

LWORK (input) INTEGER
The dimension of the array WORK. If N <= 1, LWORK must be at least 1. If JOBZ = 'N' and N > 1, LWORK must be at least 2*N+1. If JOBZ = 'V' and N > 1, LWORK must be at least 1 + 6*N + 2*N**2. If LWORK = -1, then a workspace query is assumed; the routine only calculates the optimal sizes of the WORK and IWORK arrays, returns these values as the first entries of the WORK and IWORK arrays, and no error message related to LWORK or LIWORK is issued by XERBLA.

IWORK (workspace/output) INTEGER array, dimension (MAX(1,LIWORK))
On exit, if INFO = 0, IWORK(1) returns the optimal LIWORK.

LIWORK (input) INTEGER
The dimension of the array IWORK. If N <= 1, LIWORK must be at least 1. If JOBZ = 'N' and N > 1, LIWORK must be at least 1. If JOBZ = 'V' and N > 1, LIWORK must be at least 3 + 5*N. If LIWORK = -1, then a workspace query is assumed; the routine only calculates the optimal sizes of the WORK and IWORK arrays, returns these values as the first entries of the WORK and IWORK arrays, and no error message related to LWORK or LIWORK is issued by XERBLA.

INFO (output) INTEGER
= 0: successful exit
< 0: if INFO = -i, the i-th argument had an illegal value
> 0: if INFO = i and JOBZ = 'N', then the algorithm failed to converge; i off-diagonal elements of an intermediate tridiagonal form did not converge to zero; if INFO = i and JOBZ = 'V', then the algorithm failed to compute an eigenvalue while working on the submatrix lying in rows and columns INFO/(N+1) through mod(INFO,N+1).

FURTHER DETAILS

Based on contributions by
Jeff Rutter, Computer Science Division, University of California
at Berkeley, USA
Modified by Francoise Tisseur, University of Tennessee.
Modified description of INFO. Sven, 16 Feb 05.