NAG Fortran Library

Mark 21 News

1 Introduction

At Mark 21 of the NAG Fortran Library new functionality has been introduced in addition to improvements in existing areas. The Library now contains 1533 user-callable routines, all of which are documented, and 285 are new at this mark.

A new chapter on large scale eigenproblems has been introduced, and extensions have been included in the areas of optimization, dense and banded linear algebra, direct solution of large scale linear systems, simple statistical calculations, regression, random numbers, and special functions.

The new Chapter F12 (Large Scale Eigenproblems) has routines for the solution of symmetric and nonsymmetric standard and generalized large scale eigenvalue problems. Chapter F11 has been renamed as Large Scale Linear Systems, and new routines for the direct solution of sparse problems have been added.

Chapter E04 (Minimizing or Maximizing a Function) has been updated with new routines for the solution of LP, QP and nonlinear programming problems with sparse linear constraints.

Chapter F07 (Linear Equations (LAPACK)) and Chapter F08 (Least-squares and Eigenvalue Problems (LAPACK)) have been extended to include all the LAPACK driver routines, thus allowing the solution of most problems with a call to a single routine rather than multiple calls to LAPACK computational routines. A comprehensive suite of driver routines for the solution of dense and banded linear equations has also been added to Chapter F04 (Simultaneous Linear Equations).

Routines for Landau and Vavilov distributions have been added to Chapter G01 (Simple Calculations on Statistical Data), new routines for stepwise regression and mixed effects regression have been included in Chapter G02 (Correlation and Regression Analysis), and a number of new random number generators, including Copulas and improved quasi-random number generators have been added to Chapter G05 (Random Number Generators).

Variant routines for the log Gamma function, and Bessel function of the 1st kind have been added to Chapter S (Approximations of Special Functions).

The NAG Fortran Library Manual has undergone a fundamental change since Mark 20 and the Essential Introduction is ***essential*** reading for all users of the NAG Fortran Library.

2 New Routines

The 285 new user-callable routines included in the Fortran Library at Mark 21 are as follows.

A00ACF	Check availability of a valid licence key
E04NPF	Initialization routine for E04NQF
E04NQF	LP or QP problem (suitable for sparse problems)
E04NRF	Supply optional parameter values for E04NQF from external file
E04NSF	Set a single option for E04NQF from a character string
E04NTF	Set a single option for E04NQF from an INTEGER argument
E04NUF	Set a single option for E04NQF from a ***double precision*** argument
E04NXF	Get the setting of an INTEGER valued option of E04NQF
E04NYF	Get the setting of a ***double precision*** valued option of E04NQF
E04VGF	Initialization routine for E04VHF
E04VHF	General sparse nonlinear optimizer
E04VJF	Determine the pattern of nonzeros in the Jacobian matrix for E04VHF
E04VKF	Supply optional parameter values for E04VHF from external file
E04VLF	Set a single option for E04VHF from a character string
E04VMF	Set a single option for E04VHF from an INTEGER argument
E04VNF	Set a single option for E04VHF from a ***double precision*** argument
E04VRF	Get the setting of an INTEGER valued option of E04VHF

E04VSF Get the setting of a ***double precision*** valued option of E04VHF
E04WCF Initialization routine for E04WDF
E04WDF Solves the nonlinear programming (NP) problem
E04WEF Supply optional parameter values for E04WDF from external file
E04WFF Set a single option for E04WDF from a character string
E04WGF Set a single option for E04WDF from an INTEGER argument
E04WHF Set a single option for E04WDF from a ***double precision*** argument
E04WJF Determine whether an E04WDF option has been set or not
E04WKF Get the setting of an INTEGER valued option of E04WDF
E04WLF Get the setting of a ***double precision*** valued option of E04WDF
F04BAF Computes the solution and error-bound to a real system of linear equations
F04BBF Computes the solution and error-bound to a real banded system of linear equations
F04BCF Computes the solution and error-bound to a real tridiagonal system of linear equations
F04BDF Computes the solution and error-bound to a real symmetric positive-definite system of linear equations
F04BEF Computes the solution and error-bound to a real symmetric positive-definite system of linear equations, packed storage
F04BFF Computes the solution and error-bound to a real symmetric positive-definite banded system of linear equations
F04BGF Computes the solution and error-bound to a real symmetric positive-definite tridiagonal system of linear equations
F04BHF Computes the solution and error-bound to a real symmetric system of linear equations
F04BJF Computes the solution and error-bound to a real symmetric system of linear equations, packed storage
F04CAF Computes the solution and error-bound to a complex system of linear equations
F04CBF Computes the solution and error-bound to a complex banded system of linear equations
F04CCF Computes the solution and error-bound to a complex tridiagonal system of linear equations
F04CDF Computes the solution and error-bound to a complex Hermitian positive-definite system of linear equations
F04CEF Computes the solution and error-bound to a complex Hermitian positive-definite system of linear equations, packed storage
F04CFF Computes the solution and error-bound to a complex Hermitian positive-definite banded system of linear equations
F04CGF Computes the solution and error-bound to a complex Hermitian positive-definite tridiagonal system of linear equations
F04CHF Computes the solution and error-bound to a complex Hermitian system of linear equations
F04CJF Computes the solution and error-bound to a complex Hermitian system of linear equations, packed storage
F04DHF Computes the solution and error-bound to a complex symmetric system of linear equations
F04DJF Computes the solution and error-bound to a complex symmetric system of linear equations, packed storage.
F06FEF Multiply real vector by reciprocal of scalar
F06KEF Multiply complex vector by reciprocal of real scalar
F06RNF 1-norm, ∞-norm, Frobenius norm, largest absolute element, real tridiagonal matrix
F06RPF 1-norm, ∞-norm, Frobenius norm, largest absolute element, real symmetric tridiagonal matrix
F06TAF Matrix-vector product, complex symmetric matrix
F06TBF Rank-1 update, complex symetric matrix
F06TCF Matrix-vector product, complex symmetric packed matrix
F06TDF Rank-1 update, complex symetric packed matrix
F06UNF 1-norm, ∞-norm, Frobenius norm, largest absolute element, complex tridiagonal matrix
F06UPF 1-norm, ∞-norm, Frobenius norm, largest absolute element, complex Hermitian tridiagonal matrix
F07AAF Computes the solution to a real system of linear equations
F07ABF Uses the LU factorization to compute the solution, error-bound and condition estimate for a real system of linear equations
F07AFF Computes row and column scalings intended to equilibrate a general real matrix and reduce its condition number
F07ANF Computes the solution to a complex system of linear equations

F07APF Uses the *LU* factorization to compute the solution, error-bound and condition estimate for a complex system of linear equations
F07ATF Computes row and column scalings intended to equilibrate a general complex matrix and reduce its condition number
F07BAF Computes the solution to a real banded system of linear equations
F07BBF Uses the *LU* factorization to compute the solution, error-bound and condition estimate for a real banded system of linear equations
F07BFF Computes row and column scalings intended to equilibrate a real banded matrix and reduce its condition number
F07BNF Computes the solution to a complex banded system of linear equations
F07BPF Uses the *LU* factorization to compute the solution, error-bound and condition estimate for a complex banded system of linear equations
F07BTF Computes row and column scalings intended to equilibrate a complex banded matrix and reduce its condition number
F07CAF Computes the solution to a real tridiagonal system of linear equations
F07CBF Uses the *LU* factorization to compute the solution, error-bound and condition estimate for a real tridiagonal system of linear equations
F07CDF *LU* factorization of real tridiagonal matrix
F07CEF Solves a real tridiagonal system of linear equations using the *LU* factorization computed by F07CDF (DGTTRF)
F07CGF Estimates the reciprocal of the condition number of a real tridiagonal matrix using the *LU* factorization computed by F07CDF (DGTTRF)
F07CHF Refined solution with error bounds of real tridiagonal system of linear equations, multiple right-hand sides
F07CNF Computes the solution to a complex tridiagonal system of linear equations
F07CPF Uses the *LU* factorization to compute the solution, error-bound and condition estimate for a complex tridiagonal system of linear equations
F07CRF *LU* factorization of complex tridiagonal matrix
F07CSF Solves a complex tridiagonal system of linear equations using the *LU* factorization computed by F07CDF (DGTTRF)
F07CUF Estimates the reciprocal of the condition number of a complex tridiagonal matrix using the *LU* factorization computed by F07CDF (DGTTRF)
F07CVF Refined solution with error bounds of complex tridiagonal system of linear equations, multiple right-hand sides
F07FAF Computes the solution to a real symmetric positive-definite system of linear equations
F07FBF Uses the Cholesky factorization to compute the solution, error-bound and condition estimate for a real symmetric positive-definite system of linear equations
F07FFF Computes row and column scalings intended to equilibrate a real symmetric positive-definite matrix and reduce its condition number
F07FNF Computes the solution to a complex Hermitian positive-definite system of linear equations
F07FPF Uses the Cholesky factorization to compute the solution, error-bound and condition estimate for a complex Hermitian positive-definite system of linear equations
F07FTF Computes row and column scalings intended to equilibrate a complex Hermitian positive-definite matrix and reduce its condition number
F07GAF Computes the solution to a real symmetric positive-definite system of linear equations, packed storage
F07GBF Uses the Cholesky factorization to compute the solution, error-bound and condition estimate for a real symmetric positive-definite system of linear equations, packed storage
F07GFF Computes row and column scalings intended to equilibrate a real symmetric positive-definite matrix and reduce its condition number, packed storage
F07GNF Computes the solution to a complex Hermitian positive-definite system of linear equations, packed storage
F07GPF Uses the Cholesky factorization to compute the solution, error-bound and condition estimate for a complex Hermitian positive-definite system of linear equations, packed storage
F07GTF Computes row and column scalings intended to equilibrate a complex Hermitian positive-definite matrix and reduce its condition number, packed storage
F07HAF Computes the solution to a real symmetric positive-definite banded system of linear equations
F07HBF Uses the Cholesky factorization to compute the solution, error-bound and condition estimate for a real symmetric positive-definite banded system of linear equations

F07HFF Computes row and column scalings intended to equilibrate a real symmetric positive-definite banded matrix and reduce its condition number
F07HNF Computes the solution to a complex Hermitian positive-definite banded system of linear equations
F07HPF Uses the Cholesky factorization to compute the solution, error-bound and condition estimate for a complex Hermitian positive-definite banded system of linear equations
F07HTF Computes row and column scalings intended to equilibrate a complex Hermitian positive-definite banded matrix and reduce its condition number
F07JAF Computes the solution to a real symmetric positive-definite tridiagonal system of linear equations
F07JBF Uses the modified Cholesky factorization to compute the solution, error-bound and condition estimate for a real symmetric positive-definite tridiagonal system of linear equations
F07JDF Computes the modified Cholesky factorization of a real symmetric positive-definite tridiagonal matrix
F07JEF Solves a real symmetric positive-definite tridiagonal system using the modified Cholesky factorization computed by F07JDF (DPTTRF)
F07JGF Computes the reciprocal of the condition number of a real symmetric positive-definite tridiagonal system using the modified Cholesky factorization computed by F07JDF (DPTTRF)
F07JHF Refined solution with error bounds of real symmetric positive-definite tridiagonal system of linear equations, multiple right-hand sides
F07JNF Computes the solution to a complex Hermitian positive-definite tridiagonal system of linear equations
F07JPF Uses the modified Cholesky factorization to compute the solution, error-bound and condition estimate for a complex Hermitian positive-definite tridiagonal system of linear equations
F07JRF Computes the modified Cholesky factorization of a complex Hermitian positive-definite tridiagonal matrix
F07JSF Solves a complex Hermitian positive-definite tridiagonal system using the modified Cholesky factorization computed by F07JRF (ZPTTRF)
F07JUF Computes the reciprocal of the condition number of a complex Hermitian positive-definite tridiagonal system using the modified Cholesky factorization computed by F07JRF (ZPTTRF)
F07JVF Refined solution with error bounds of complex Hermitian positive-definite tridiagonal system of linear equations, multiple right-hand sides
F07MAF Computes the solution to a real symmetric system of linear equations
F07MBF Uses the diagonal pivoting factorization to compute the solution to a real symmetric system of linear equations
F07MNF Computes the solution to a complex Hermitian system of linear equations
F07MPF Uses the diagonal pivoting factorization to compute the solution to a complex Hermitian system of linear equations
F07NNF Computes the solution to a complex symmetric system of linear equations
F07NPF Uses the diagonal pivoting factorization to compute the solution to a complex symmetric system of linear equations
F07PAF Computes the solution to a real symmetric system of linear equations, packed storage
F07PBF Uses the diagonal pivoting factorization to compute the solution to a real symmetric system of linear equations, packed storage
F07PNF Computes the solution to a complex Hermitian system of linear equations, packed storage
F07PPF Uses the diagonal pivoting factorization to compute the solution to a complex Hermitian system of linear equations, packed storage
F07QNF Computes the solution to a complex symmetric system of linear equations, packed storage
F07QPF Uses the diagonal pivoting factorization to compute the solution to a complex symmetric system of linear equations, packed storage
F08AAF Solves an overdetermined or underdetermined real linear system
F08ANF Solves an overdetermined or underdetermined complex linear system
F08BAF Computes the minimum-norm solution to a real linear least-squares problem
F08BFF *QR* factorization of real general rectangular matrix with column pivoting, using BLAS-3
F08BHF Reduces a real upper trapezoidal matrix to upper triangular form
F08BKF Apply orthogonal transformation determined by F08BHF (DTZRZF)
F08BNF Computes the minimum-norm solution to a complex linear least-squares problem
F08BTF *QR* factorization of complex general rectangular matrix with column pivoting, using BLAS-3
F08BVF Reduces a complex upper trapezoidal matrix to upper triangular form

F08BXF Apply unitary transformation determined by F08BVF (ZTZRZF)
F08CEF *QL* factorization of real general rectangular matrix
F08CFF Form all or part of orthogonal *Q* from *QL* factorization determined by F08CEF (DGEQLF)
F08CGF Apply orthogonal transformation determined by F08CEF (DGEQLF)
F08CHF *RQ* factorization of real general rectangular matrix
F08CJF Form all or part of orthogonal *Q* from *RQ* factorization determined by F08CHF (DGERQF)
F08CKF Apply orthogonal transformation determined by F08CHF (DGERQF)
F08CSF *QL* factorization of complex general rectangular matrix
F08CTF Form all or part of orthogonal *Q* from *QL* factorization determined by F08CSF (ZGEQLF)
F08CUF Apply unitary transformation determined by F08CSF (ZGEQLF)
F08CVF *RQ* factorization of complex general rectangular matrix
F08CWF Form all or part of orthogonal *Q* from *RQ* factorization determined by F08CVF (ZGERQF)
F08CXF Apply unitary transformation determined by F08CVF (ZGERQF)
F08FAF Computes all eigenvalues and, optionally, eigenvectors of a real symmetric matrix
F08FBF Computes selected eigenvalues and, optionally, eigenvectors of a real symmetric matrix
F08FDF Computes selected eigenvalues and, optionally, eigenvectors of a real symmetric matrix (Relatively Robust Representations)
F08FLF Computes the reciprocal condition numbers for the eigenvectors of a real symmetric or complex Hermitian matrix or for the left or right singular vectors of a general matrix
F08FNF Computes all eigenvalues and, optionally, eigenvectors of a complex Hermitian matrix
F08FPF Computes selected eigenvalues and, optionally, eigenvectors of a complex Hermitian matrix
F08FRF Computes selected eigenvalues and, optionally, eigenvectors of a complex Hermitian matrix (Relatively Robust Representations)
F08GAF Computes all eigenvalues and, optionally, eigenvectors of a real symmetric matrix, packed storage
F08GBF Computes selected eigenvalues and, optionally, eigenvectors of a real symmetric matrix, packed storage
F08GNF Computes all eigenvalues and, optionally, eigenvectors of a complex Hermitian matrix, packed storage
F08GPF Computes selected eigenvalues and, optionally, eigenvectors of a complex Hermitian matrix, packed storage
F08HAF Computes all eigenvalues and, optionally, eigenvectors of a real symmetric band matrix
F08HBF Computes selected eigenvalues and, optionally, eigenvectors of a real symmetric band matrix
F08HNF Computes all eigenvalues and, optionally, eigenvectors of a complex Hermitian band matrix
F08HPF Computes selected eigenvalues and, optionally, eigenvectors of a complex Hermitian band matrix
F08JAF Computes all eigenvalues and, optionally, eigenvectors of a real symmetric tridiagonal matrix
F08JBF Computes selected eigenvalues and, optionally, eigenvectors of a real symmetric tridiagonal matrix
F08JDF Computes selected eigenvalues and, optionally, eigenvectors of a real symmetric tridiagonal matrix (Relatively Robust Representations)
F08JHF Computes all eigenvalues and, optionally, eigenvectors of a real symmetric tridiagonal matrix or a matrix reduced to this form (divide-and-conquer)
F08JLF Computes all eigenvalues and, optionally, eigenvectors of a real symmetric tridiagonal matrix or a symmetric matrix reduced to this form (Relatively Robust Representations)
F08JVF Computes all eigenvalues and, optionally, eigenvectors of a real symmetric tridiagonal matrix or a complex Hermitian matrix reduced to this form (divide-and-conquer)
F08JYF Computes all eigenvalues and, optionally, eigenvectors of a real symmetric tridiagonal matrix or a complex Hermitian matrix reduced to this form (Relatively Robust Representations)
F08KAF Computes the minimum-norm solution to a real linear least-squares problem using singular value decomposition
F08KBF Computes the singular value decomposition of a real matrix, optionally computing the left and/or right singular vectors
F08KCF Computes the minimum-norm solution to a real linear least-squares problem using singular value decomposition (divide-and-conquer)
F08KDF Computes the singular value decomposition of a real matrix, optionally computing the left and/or right singular vectors (divide-and-conquer)
F08KNF Computes the minimum-norm solution to a complex linear least-squares problem using singular value decomposition

F08KPF Computes the singular value decomposition of a complex matrix, optionally computing the left and/or right singular vectors

F08KQF Computes the minimum-norm solution to a complex linear least-squares problem using singular value decomposition (divide-and-conquer)

F08KRF Computes the singular value decomposition of a complex matrix, optionally computing the left and/or right singular vectors (divide-and-conquer)

F08MDF Computes the singular value decomposition of a real bidiagonal matrix, optionally computing the singular vectors (divide-and-conquer)

F08NAF Computes all eigenvalues and, optionally, left and/or right eigenvectors of a real nonsymmetric matrix

F08NBF Computes all eigenvalues and, optionally, left and/or right eigenvectors of a real nonsymmetric matrix; also, optionally, the balancing transformation, the reciprocal condition numbers for the eigenvalues and for the right eigenvectors

F08NNF Computes all eigenvalues and, optionally, left and/or right eigenvectors of a complex nonsymmetric matrix

F08NPF Computes all eigenvalues and, optionally, left and/or right eigenvectors of a complex nonsymmetric matrix; also, optionally, the balancing transformation, the reciprocal condition numbers for the eigenvalues and for the right eigenvectors

F08PAF Computes for real square nonsymmetric matrix, the eigenvalues, the real Schur form, and, optionally, the matrix of Schur vectors

F08PBF Computes for real square nonsymmetric matrix, the eigenvalues, the real Schur form, and, optionally, the matrix of Schur vectors; also, optionally, computes reciprocal condition numbers for selected eigenvalues

F08PNF Computes for complex square nonsymmetric matrix, the eigenvalues, the Schur form, and, optionally, the matrix of Schur vectors

F08PPF Computes for real square nonsymmetric matrix, the eigenvalues, the Schur form, and, optionally, the matrix of Schur vectors; also, optionally, computes reciprocal condition numbers for selected eigenvalues

F08SAF Computes all the eigenvalues, and optionally, the eigenvectors of a real generalized symmetric-definite eigenproblem

F08SBF Computes selected eigenvalues, and optionally, the eigenvectors of a real generalized symmetric-definite eigenproblem

F08SCF Computes all the eigenvalues, and optionally, the eigenvectors of a real generalized symmetric-definite eigenproblem (divide-and-conquer)

F08SNF Computes all the eigenvalues, and optionally, the eigenvectors of a complex generalized Hermitian-definite eigenproblem

F08SPF Computes selected eigenvalues, and optionally, the eigenvectors of a complex generalized Hermitian-definite eigenproblem

F08SQF Computes all the eigenvalues, and optionally, the eigenvectors of a complex generalized Hermitian-definite eigenproblem (divide-and-conquer)

F08TAF Computes all the eigenvalues, and optionally, the eigenvectors of a real generalized symmetric-definite eigenproblem, packed storage

F08TBF Computes selected eigenvalues, and optionally, the eigenvectors of a real generalized symmetric-definite eigenproblem, packed storage

F08TCF Computes all the eigenvalues, and optionally, the eigenvectors of a real generalized symmetric-definite eigenproblem, packed storage (divide-and-conquer)

F08TNF Computes all the eigenvalues, and optionally, the eigenvectors of a complex generalized Hermitian-definite eigenproblem, packed storage

F08TPF Computes selected eigenvalues, and optionally, the eigenvectors of a complex generalized Hermitian-definite eigenproblem, packed storage

F08TQF Computes selected eigenvalues, and optionally, the eigenvectors of a complex generalized Hermitian-definite eigenproblem, packed storage (divide-and-conquer)

F08UAF Computes all the eigenvalues, and optionally, the eigenvectors of a real banded generalized symmetric-definite eigenproblem

F08UBF Computes selected eigenvalues, and optionally, the eigenvectors of a real banded generalized symmetric-definite eigenproblem

F08UCF Computes all the eigenvalues, and optionally, the eigenvectors of a real banded generalized symmetric-definite eigenproblem (divide-and-conquer)

F08UNF Computes all the eigenvalues, and optionally, the eigenvectors of a complex banded generalized Hermitian-definite eigenproblem
F08UPF Computes selected eigenvalues, and optionally, the eigenvectors of a complex banded generalized Hermitian-definite eigenproblem
F08UQF Computes all the eigenvalues, and optionally, the eigenvectors of a complex banded generalized Hermitian-definite eigenproblem (divide-and-conquer)
F08VAF Computes the generalized singular value decomposition of a real matrix pair
F08VEF Computes orthogonal matrices as processing steps for computing the generalized singular value decomposition of a real matrix pair
F08VNF Computes the generalized singular value decomposition of a complex matrix pair
F08VSF Computes orthogonal matrices as processing steps for computing the generalized singular value decomposition of a complex matrix pair
F08WAF Computes, for a real nonsymmetric matrix pair, the generalized eigenvalues, and optionally, the left and/or right generalized eigenvectors
F08WBF Computes, for a real nonsymmetric matrix pair, the generalized eigenvalues, and optionally, the left and/or right generalized eigenvectors; also, optionally, the balancing transformation, the reciprocal condition numbers for the eigenvalues and for the right eigenvectors
F08WNF Computes, for a complex nonsymmetric matrix pair, the generalized eigenvalues, and optionally, the left and/or right generalized eigenvectors
F08WPF Computes, for a complex nonsymmetric matrix pair, the generalized eigenvalues, and optionally, the left and/or right generalized eigenvectors; also, optionally, the balancing transformation, the reciprocal condition numbers for the eigenvalues and for the right eigenvectors
F08XAF Computes, for a real nonsymmetric matrix pair, the generalized eigenvalues, the generalized real Schur form and, optionally, the left and/or right matrices of Schur vectors
F08XBF Computes, for a real nonsymmetric matrix pair, the generalized eigenvalues, the generalized real Schur form and, optionally, the left and/or right matrices of Schur vectors; also, optionally, computes reciprocal condition numbers for selected eigenvalues
F08XNF Computes, for a complex nonsymmetric matrix pair, the generalized eigenvalues, the generalized complex Schur form and, optionally, the left and/or right matrices of Schur vectors
F08XPF Computes, for a complex nonsymmetric matrix pair, the generalized eigenvalues, the generalized complex Schur form and, optionally, the left and/or right matrices of Schur vectors; also, optionally, computes reciprocal condition numbers for selected eigenvalues
F08YEF Computes the generalized singular value decomposition of a real upper triangular (or trapezoidal) matrix pair
F08YFF Reorders the generalized real Schur decomposition of a real matrix pair using an orthogonal equivalence transformation
F08YGF Reorders the generalized real Schur decomposition of a real matrix pair using an orthogonal equivalence transformation, computes the generalized eigenvalues of the reordered pair and, optionally, computes the estimates of reciprocal condition numbers for eigenvalues and eigenspaces
F08YHF Solves the real-valued generalized Sylvester equation
F08YLF Estimates reciprocal condition numbers for specified eigenvalues and/or eigenvectors of a real matrix pair in generalized real Schur canonical form
F08YSF Computes the generalized singular value decomposition of a complex upper triangular (or trapezoidal) matrix pair
F08YTF Reorders the generalized Schur decomposition of a complex matrix pair using an unitary equivalence transformation
F08YUF Reorders the generalized Schur decomposition of a complex matrix pair using an unitary equivalence transformation, computes the generalized eigenvalues of the reordered pair and, optionally, computes the estimates of reciprocal condition numbers for eigenvalues and eigenspaces
F08YVF Solves the complex generalized Sylvester equation
F08YYF Estimates reciprocal condition numbers for specified eigenvalues and/or eigenvectors of a complex matrix pair in generalized Schur canonical form
F08ZAF Solves the real linear equality-constrained least-squares (LSE) problem
F08ZBF Solves a real general Gauss–Markov linear model (GLM) problem
F08ZEF Computes a generalized QR factorization of a real matrix pair
F08ZFF Computes a generalized RQ factorization of a real matrix pair

F08ZNF	Solves the complex linear equality-constrained least-squares (LSE) problem
F08ZPF	Solves a complex general Gauss–Markov linear model (GLM) problem
F08ZSF	Computes a generalized QR factorization of a complex matrix pair
F08ZTF	Computes a generalized RQ factorization of a complex matrix pair
F11MDF	Real sparse nonsymmetric linear systems, setup for F11MEF
F11MEF	LU factorization of real sparse matrix
F11MFF	Solution of real sparse simultaneous linear equations (coefficient matrix already factorized)
F11MGF	Estimate condition number of real matrix, matrix already factorized by F11MEF
F11MHF	Refined solution with error bounds of real system of linear equations, multiple right-hand sides
F11MKF	Real sparse nonsymmetric matrix matrix multiply, compressed column storage
F11MLF	1-norm, ∞-norm, largest absolute element, real general matrix
F11MMF	Real sparse nonsymmetric linear systems, diagnostic for F11MEF
F12AAF	Initialization routine for (F12ABF) computing selected eigenvalues and, optionally, eigenvectors of a real nonsymmetric sparse (standard or generalized) eigenproblem
F12ABF	Implements a reverse communication interface for the Implicitly Restarted Arnoldi iteration for computing selected eigenvalues and, optionally, eigenvectors of a real nonsymmetric sparse (standard or generalized) eigenproblem
F12ACF	Returns the converged approximations (as determined by F12ABF) to eigenvalues of a real nonsymmetric sparse (standard or generalized) eigenproblem and, optionally, the corresponding approximate eigenvectors and/or an orthonormal basis for the associated approximate invariant subspace
F12ADF	Set a single option from a string (F12ABF/F12ACF/F12AGF)
F12AEF	Provides monitoring information for F12ABF
F12AFF	Initialization routine for (F12AGF) computing selected eigenvalues and, optionally, eigenvectors of a real nonsymmetric banded (standard or generalized) eigenproblem
F12AGF	Computes approximations to selected eigenvalues of a real nonsymmetric banded (standard or generalized) eigenproblem and, optionally, the corresponding approximate eigenvectors and/or an orthonormal basis for the associated approximate invariant subspace
F12ANF	Initialization routine for (F12APF) computing selected eigenvalues and, optionally, eigenvectors of a complex sparse (standard or generalized) eigenproblem
F12APF	Implements a reverse communication interface for the Implicitly Restarted Arnoldi iteration for computing selected eigenvalues and, optionally, eigenvectors of a complex sparse (standard or generalized) eigenproblem
F12AQF	Returns the converged approximations (as determined by F12ABF) to eigenvalues of a complex sparse (standard or generalized) eigenproblem and, optionally, the corresponding approximate eigenvectors and/or an orthonormal basis for the associated approximate invariant subspace
F12ARF	Set a single option from a string (F12APF/F12AQF)
F12ASF	Provides monitoring information for F12APF
F12FAF	Initialization routine for (F12FBF) computing selected eigenvalues and, optionally, eigenvectors of a real symmetric sparse (standard or generalized) eigenproblem
F12FBF	Implements a reverse communication interface for the Implicitly Restarted Arnoldi iteration for computing selected eigenvalues and, optionally, eigenvectors of a real symmetric sparse (standard or generalized) eigenproblem
F12FCF	Returns the converged approximations (as determined by F12ABF) to eigenvalues of a real symmetric sparse (standard or generalized) eigenproblem and, optionally, the corresponding approximate eigenvectors and/or an orthonormal basis for the associated approximate invariant subspace
F12FDF	Set a single option from a string (F12FBF/F12FCF/F12FGF)
F12FEF	Provides monitoring information for F12FBF
F12FFF	Initialization routine for (F12FGF) computing selected eigenvalues and, optionally, eigenvectors of a real symmetric banded (standard or generalized) eigenproblem
F12FGF	Computes approximations to selected eigenvalues of a real symmetric banded (standard or generalized) eigenproblem and, optionally, the corresponding approximate eigenvectors and/or an orthonormal basis for the associated approximate invariant subspace
G01ETF	Landau distribution function $\Phi(\lambda)$
G01EUF	Vavilov distribution function $\Phi_V\left(\lambda;\kappa,\beta^2\right)$
G01FTF	Landau inverse function $\Psi(x)$

G01MTF	Landau density function $\phi(\lambda)$
G01MUF	Vavilov density function $\phi_V(\lambda;\kappa,\beta^2)$
G01PTF	Landau first moment function $\Phi_1(x)$
G01QTF	Landau second moment function $\Phi_2(x)$
G01RTF	Landau derivative function $\phi\prime(\lambda)$
G01ZUF	Initialization routine for G01MUF and G01EUF
G02EFF	Stepwise linear regression
G02JAF	Linear mixed effects regression using Restricted Maximum Likelihood (REML)
G02JBF	Linear mixed effects regression using Maximum Likelihood (ML)
G05LXF	Generates a matrix of random numbers from a multivariate Student's *t*-distribution, seeds and generator passed explicitly
G05LYF	Generates a matrix of random numbers from a multivariate Normal distribution, seeds and generator passed explicitly
G05RAF	Generates a matrix of random numbers from a Gaussian Copula, seeds and generator passed explicitly
G05RBF	Generates a matrix of random numbers from a Student's *t*-Copula, seeds and generator passed explicitly
G05YCF	Initializes the Faure generator (G05YDF/G05YJF/G05YKF)
G05YDF	Generates a sequence of quasi-random numbers using Faure's method
G05YEF	Initializes the Sobol generator (G05YFF/G05YJF/G05YKF)
G05YFF	Generates a sequence of quasi-random numbers using Sobol's method
G05YGF	Initializes the Neiderreiter generator (G05YHF/G05YJF/G05YKF)
G05YHF	Generates a sequence of quasi-random numbers using Neiderreiter's method
G05YJF	Generates a Normal quasi-random number sequence using Faure's, Sobol's or Neiderreiter's method
G05YKF	Generates a log-Normal quasi-random number sequence using Faure's, Sobol's or Neiderreiter's method
S14AGF	Logarithm of the Gamma function $\ln\Gamma(z)$
S18GKF	Bessel function of the 1st kind $J_{\alpha\pm n}(z)$

3 Withdrawn Routines

The following routines have been withdrawn from the NAG Fortran Library at Mark 21. Warning of their withdrawal was included in the NAG Fortran Library Manual at Mark 20, together with advice on which routines to use instead. See the document 'Advice on Replacement Calls for Withdrawn/Superseded Routines' for more detailed guidance.

Withdrawn Routine	**Replacement Routine(s)**
F11BAF	F11BDF
F11BBF	F11BEF
F11BCF	F11BFF

4 Routines Scheduled for Withdrawal

The routines listed below are scheduled for withdrawal from the Fortran Library, because improved routines have now been included in the Library. Users are advised to stop using routines which are scheduled for withdrawal immediately and to use recommended replacement routines instead. See the document 'Advice on Replacement Calls for Withdrawn/Superseded Routines' for more detailed guidance, including advice on how to change a call to the old routine into a call to its recommended replacement.

The following routines will be withdrawn at Mark 22.

Routine Scheduled for Withdrawal	**Replacement Routine(s)**
E04UNF	E04USF/E04USA
F11GAF	F11GDF
F11GBF	F11GEF

F11GCF	F11GFF
G05CAF	G05KAF
G05CBF	G05KBF
G05CCF	G05KCF
G05CFF	F06DFF
G05CGF	F06DFF
G05DAF	G05LGF
G05DBF	G05LJF
G05DCF	G05LNF
G05DDF	G05LAF
G05DEF	G05LKF
G05DFF	G05LLF
G05DHF	G05LCF
G05DJF	G05LBF
G05DKF	G05LDF
G05DPF	G05LMF
G05DRF	G05MEF
G05DYF	G05MAF
G05DZF	G05KEF
G05EAF	G05LZF
G05EBF	G05MAF
G05ECF	G05MKF
G05EDF	G05MJF
G05EEF	G05MCF
G05EFF	G05MLF
G05EGF	G05PAF
G05EHF	G05NAF
G05EJF	G05NBF
G05EWF	G05PAF
G05EXF	G05MZF
G05EYF	G05MZF
G05EZF	G05LZF
G05FAF	G05LGF
G05FBF	G05LJF
G05FDF	G05LAF
G05FEF	G05LEF
G05FFF	G05LFF
G05FSF	G05LPF
G05GAF	G05QAF
G05GBF	G05QBF
G05HDF	G05PCF
G05ZAF	No replacement document required

The following routines have been superseded, but will not be withdrawn from the Library until Mark 23 at the earliest.

Superseded Routine	**Replacement Routine(s)**
E04NKF/E04NKA	E04NQF
E04NLF/E04NLA	E04NRF
E04NMF/E04NMA	E04NSF, E04NTF and E04NUF
E04UCF/E04UCA	E04WDF
F02BJF	F08WAF (DGGEV)
F02EAF	F08PAF (DGEES)
F02EBF	F08NAF (DGEEV)
F02FAF	F08FAF (DSYEV)
F02FCF	F08FBF (DSYEVX)
F02FDF	F08SAF (DSYGV)
F02FHF	F08UAF (DSBGV)

F02GAF	F08PNF (ZGEES)
F02GBF	F08NNF (ZGEEV)
F02GJF	F08WNF (ZGGEV)
F02HAF	F08FNF (ZHEEV)
F02HCF	F08FPF (ZHEEVX)
F02HDF	F08SNF (ZHEGV)
F02WEF	F08KBF (DGESVD)
F02XEF	F08KPF (ZGESVD)
F04AAF	F07AAF (DGESV)
F04ACF	F07HAF (DPBSV)
F04ADF	F07ANF (ZGESV)
F04ARF	F07AAF (DGESV)
F04EAF	F07CAF (DGTSV)
F04FAF	F07JAF (DPTSV), F07JDF (DPTTRF) and F07JEF (DPTTRS)
F04JAF	F08KAF (DGELSS)
F04JDF	F08KAF (DGELSS)
F04JLF	F08ZBF (DGGGLM)
F04JMF	F08ZAF (DGGLSE)
F04KLF	F08ZPF (ZGGGLM)
F04KMF	F08ZNF (ZGGLSE)
G05YAF	G05YCF, G05YDF, G05YEF, G05YFF, G05YGF, G05YHF, G05YJF and G05YKF
G05YBF	G05YCF, G05YDF, G05YEF, G05YFF, G05YGF, G05YHF, G05YJF and G05YKF