NAG Library

NAG C Library News, Mark 24

1 Introduction

At Mark 24 of the NAG C Library new functionality has been introduced in addition to improvements in existing areas. The Library now contains 1516 user-callable functions, all of which are documented, of which 148 are new at this mark.

A major feature at this mark is the incorporation of SMP capability into the C Library. Consequently the C Library documentation is being enhanced to indicate which functions are capable of utilizing threads.

A new Chapter x07 (IEEE Arithmetic) has been introduced, providing functions relating to IEEE arithmetic such as determining or creating an infinite value or a NaN (Not a Number). There have also been extensions in functionality included in the areas of statistics, wavelets, ordinary differential equations, interpolation, surface fitting, optimization, matrix operations, linear algebra, operations research, and special functions.

Chapter c06 (Fourier Transforms) has Fast Fourier Transforms (FFTs) for two-dimensional and three-dimensional real data, and replacement functions for transforms of complex data and symmetric transforms of real data removing restrictions on sequence length.

Chapter c09 (Wavelet Transforms) has new functions for computing the three-dimensional discrete wavelet transform (DWT) and its inverse, including applying the DWT over multiple levels (c09f*c). These functions are threaded for parallelism in some implementations. In addition there are new functions for inserting coefficients into and extracting coefficients from the compact form used in the multilevel two-dimensional functions and all three-dimensional functions, which would make working with the DWT functions easier as demonstrated in the examples. There are also functions for the maximal overlap discrete wavelet transform (MODWT) and its inverse in one dimension, which are useful in time series analysis.

Chapter d01 (Quadrature) has a comprehensive reverse communication one-dimensional adaptive quadrature function and a variant for badly behaved integrands.

Chapter d02 (Ordinary Differential Equations) has a suite of functions solving Boundary Value Problems by the collocation method, and a replacement for the suite implementing Runge–Kutta methods for non-stiff Initial Value Problems.

Chapter e01 (Interpolation) has the modified Shepard's method for interpolating in dimensions greater than 5.

Chapter e02 (Curve and Surface Fitting) has a two-stage approximation method for two-dimensional scattered data.

Chapter e04 (Minimizing or Maximizing a Function) has non-negative least squares and an improved MPS data reader.

Chapter e05 (Global Optimization of a Function) has a multi-start version of a least squares with nonlinear constraints function.

Chapter f01 (Matrix Operations, Including Inversion) has greatly extended its range of matrix functions. New and improved algorithms are available for the matrix logarithm, the matrix square root, the matrix exponential and general matrix powers. We also now have functions for computing Fréchet derivatives.

Chapter f02 (Eigenvalues and Eigenvectors) has functions for the solution of the real and complex quadratic eigenvalue problem, and a driver function for calculating selected eigenvalues/vectors of general sparse matrices.

Chapter f04 (Simultaneous Linear Equations) has norm estimators for rectangular matrices.

Chapter f08 (Least Squares and Eigenvalue Problems (LAPACK)) has functions for recursive, explicitly blocked QR factorization and applying Q for general matrices and for triangular-pentagonal matrices, and has functions for the complete CS decomposition of an orthogonal or unitary matrix.

Chapter f11 (Large Scale Linear Systems) has a block diagonal (possibly overlapping) preconditioner and associated solver for real and complex nonsymmetric sparse matrices.

Chapter f12 (Large Scale Eigenproblems) has a driver for selected eigenvalues/vectors of general banded complex eigenproblems.

Chapter f16 (NAG Interface to BLAS) has two additions from the BLAST set of functions.

Chapter g01 (Simple Calculations on Statistical Data) has functions for calculating summary statistics in a rolling window and combining summary statistics calculated on different data streams as well as a function for calculating probabilities from a multivariate Student's t-distribution.

Chapter g02 (Correlation and Regression Analysis) has functions for weighted nearest correlation matrix and calculating sums of squares matrices (and hence correlation and covariance matrices) on streamed data.

Chapter g03 (Multivariate Methods) has a Gaussian mixture model function.

Chapter g05 (Random Number Generators) has Brownian bridge and random field functions.

Chapter g13 (Time Series Analysis) has functions for analysing inhomogeneous time series.

Chapter h (Operations Research) has functions for feature subset selections.

Chapter s (Approximations of Special Functions) has confluent and Gauss hypergeometric functions ${}_1F_1(a,b;x)$ and ${}_2F_1(a,b;c;x)$, and a new derivative pricing function extends the Heston stochastic volatility model to incorporate a term structure which can facilitate calibration to market data.

2 New Functions

The 148 new user-callable functions included in the NAG C Library at Mark 24 are as follows.

Function Name	**Purpose**
c06fkc	Circular convolution or correlation of two real vectors, no restrictions on n
c06pac	Single one-dimensional real and Hermitian complex discrete Fourier transform, using complex storage format for Hermitian sequences
c06pcc	Single one-dimensional complex discrete Fourier transform, complex data type
c06psc	Multiple one-dimensional complex discrete Fourier transforms, complex data type
c06puc	Two-dimensional complex discrete Fourier transform, complex data type
c06pvc	Two-dimensional real-to-complex discrete Fourier transform
c06pwc	Two-dimensional complex-to-real discrete Fourier transform
c06pyc	Three-dimensional real-to-complex discrete Fourier transform
c06pzc	Three-dimensional complex-to-real discrete Fourier transform
c06rec	Multiple discrete sine transforms, simple
c06rfc	Multiple discrete cosine transforms, simple
c06rgc	Multiple discrete quarter-wave sine transforms, simple
c06rhc	Multiple discrete quarter-wave cosine transforms, simple
c09acc	Three-dimensional wavelet filter initialization
c09dac	One-dimensional maximal overlap discrete wavelet transform (MODWT)
c09dbc	One-dimensional inverse maximal overlap discrete wavelet transform (IMODWT)

c09dcc	One-dimensional multi-level maximal overlap discrete wavelet transform (MODWT)
c09ddc	One-dimensional inverse multi-level maximal overlap discrete wavelet transform (IMODWT)
c09eyc	Two-dimensional discrete wavelet transform coefficient extraction
c09ezc	Two-dimensional discrete wavelet transform coefficient insertion
c09fac	Three-dimensional discrete wavelet transform
c09fbc	Three-dimensional inverse discrete wavelet transform
c09fcc	Three-dimensional multi-level discrete wavelet transform
c09fdc	Three-dimensional inverse multi-level discrete wavelet transform
c09fyc	Three-dimensional discrete wavelet transform coefficient extraction
c09fzc	Three-dimensional discrete wavelet transform coefficient insertion
d01rac	One-dimensional quadrature, adaptive, finite interval, multiple integrands, vectorized abscissae, reverse communication
d01rcc	Determine required array dimensions for nag_quad_1d_gen_vec_multi_rcomm (d01rac)
d01rgc	One-dimensional quadrature, adaptive, finite interval, strategy due to Gonnet, allowing for badly behaved integrands
d01uac	One-dimensional Gaussian quadrature, choice of weight functions (vectorized)
d01zkc	Option setting function
d01zlc	Option getting function
d02pec	Ordinary differential equations, initial value problem, Runge–Kutta method, integration over range with output
d02pfc	Ordinary differential equations, initial value problem, Runge–Kutta method, integration over one step
d02pqc	Ordinary differential equations, initial value problem, setup for nag_ode_ivp_rkts_range (d02pec) and nag_ode_ivp_rkts_onestep (d02pfc)
d02prc	Ordinary differential equations, initial value problem, resets end of range for nag_ode_ivp_rkts_onestep (d02pfc)
d02psc	Ordinary differential equations, initial value problem, interpolation for nag_ode_ivp_rkts_onestep (d02pfc)
d02ptc	Ordinary differential equations, initial value problem, integration diagnostics for nag_ode_ivp_rkts_range (d02pec) and nag_ode_ivp_rkts_onestep (d02pfc)
d02puc	Ordinary differential equations, initial value problem, error assessment diagnostics for nag_ode_ivp_rkts_range (d02pec) and nag_ode_ivp_rkts_onestep (d02pfc)
d02tlc	Ordinary differential equations, general nonlinear boundary value problem, collocation technique
d02tvc	Ordinary differential equations, general nonlinear boundary value problem, setup for nag_ode_bvp_coll_nlin_solve (d02tlc)
d02txc	Ordinary differential equations, general nonlinear boundary value problem, continuation facility for nag_ode_bvp_coll_nlin_solve (d02tlc)
d02tyc	Ordinary differential equations, general nonlinear boundary value problem, interpolation for nag_ode_bvp_coll_nlin_solve (d02tlc)
d02tzc	Ordinary differential equations, general nonlinear boundary value problem, diagnostics for nag_ode_bvp_coll_nlin_solve (d02tlc)
e01zmc	Interpolating function, modified Shepard's method, d dimensions

e01znc	Interpolated values, evaluate interpolant computed by nag_nd_shep_interp (e01zmc), function and first derivatives, d dimensions
e02alc	Minimax curve fit by polynomials
e02bfc	Evaluation of fitted cubic spline, function and optionally derivatives at a vector of points
e02jdc	Spline approximation to a set of scattered data using a two-stage approximation method
e02jec	Evaluation at a vector of points of a spline computed by nag_2d_spline_fit_ts_scat (e02jdc)
e02jfc	Evaluation at a mesh of points of a spline computed by nag_2d_spline_fit_ts_scat (e02jdc)
e02zkc	Option setting routine
e02zlc	Option getting routine
e04mxc	Reads MPS data file defining LP, QP, MILP or MIQP problem
e04pcc	Computes the least squares solution to a set of linear equations subject to fixed upper and lower bounds on the variables. An option is provided to return a minimal length solution if a solution is not unique
e05usc	Global optimization of a sum of squares problem using multi-start, nonlinear constraints
f01elc	Function of a real matrix (using numerical differentiation)
f01enc	Real matrix square root
f01epc	Real upper quasi-triangular matrix square root
f01eqc	General power of a real matrix
f01flc	Function of a complex matrix (using numerical differentiation)
f01fnc	Complex matrix square root
f01fpc	Complex upper triangular matrix square root
f01fqc	General power of a complex matrix
f01gac	Action of a real matrix exponential on a real matrix
f01gbc	Action of a real matrix exponential on a real matrix (reverse communication)
f01hac	Action of a complex matrix exponential on a complex matrix
f01hbc	Action of a complex matrix exponential on a complex matrix (reverse communication)
f01jac	Condition number for the exponential, logarithm, sine, cosine, sinh or cosh of a real matrix
f01jbc	Condition number for a function of a real matrix (using numerical differentiation)
f01jcc	Condition number for a function of a real matrix (using user-supplied derivatives)
f01jdc	Condition number for square root of real matrix
f01jec	Condition number for real matrix power
f01jfc	Fréchet derivative of real matrix power
f01jgc	Condition number for real matrix exponential
f01jhc	Fréchet derivative of real matrix exponential
f01jjc	Condition number for real matrix logarithm
f01jkc	Fréchet derivative of real matrix logarithm
f01kac	Condition number for the exponential, logarithm, sine, cosine, sinh or cosh of a complex matrix
f01kbc	Condition number for a function of a complex matrix (using numerical differentiation)

f01kcc	Condition number for a function of a complex matrix (using user-supplied derivatives)
f01kdc	Condition number for square root of complex matrix
f01kec	Condition number for complex matrix power
f01kfc	Fréchet derivative of complex matrix power
f01kgc	Condition number for complex matrix exponential
f01khc	Fréchet derivative of complex matrix exponential
f01kjc	Condition number for complex matrix logarithm
f01kkc	Fréchet derivative of complex matrix logarithm
f02ekc	Selected eigenvalues and eigenvectors of a real sparse general matrix
f02jcc	Solves the quadratic eigenvalue problem for real matrices
f02jqc	Solves the quadratic eigenvalue problem for complex matrices
f04ydc	Norm estimation (for use in condition estimation), real rectangular matrix
f04zdc	Norm estimation (for use in condition estimation), complex rectangular matrix
f08abc	Performs a QR factorization of real general rectangular matrix, with explicit blocking
f08acc	Applies the orthogonal transformation determined by nag_dgeqrt (f08abc)
f08apc	Performs a QR factorization of complex general rectangular matrix using recursive algorithm
f08aqc	Applies the unitary transformation determined by nag_zgeqrt (f08apc)
f08bbc	QR factorization of real general triangular-pentagonal matrix
f08bcc	Applies the orthogonal transformation determined by nag_dtpqrt (f08bbc)
f08bpc	QR factorization of complex triangular-pentagonal matrix
f08bqc	Applies the unitary transformation determined by nag_ztpqrt (f08bpc)
f08rac	Computes the CS decomposition of an orthogonal matrix partitioned into four real submatrices
f08rnc	Computes the CS decomposition of an unitary matrix partitioned into four complex submatrices
f11dfc	Real sparse nonsymmetric linear system, incomplete LU factorization of local or overlapping diagonal blocks
f11dgc	Solution of real sparse nonsymmetric linear system, RGMRES, CGS, Bi-CGSTAB or TFQMR method, incomplete LU block diagonal preconditioner computed by nag_sparse_nsym_precon_bdilu (f11dfc)
f11dtc	Complex, sparse, non-Hermitian linear system, incomplete LU factorization of local or overlapping diagonal blocks
f11duc	Solution of complex, sparse, non-Hermitian linear system, RGMRES, CGS, Bi-CGSTAB or TFQMR method, incomplete LU block diagonal preconditioner computed by nag_sparse_nherm_precon_bdilu (f11dtc)
f12atc	Initialization function for nag_complex_banded_eigensystem_solve (f12auc) computing selected eigenvalues and, optionally, eigenvectors of a complex banded (standard or generalized) eigenproblem.
f12auc	Selected eigenvalues and, optionally, eigenvectors of complex non-Hermitian banded eigenproblem, driver
f16eac	Dot product of two vectors, allows scaling and accumulation.
f16gcc	Complex weighted vector addition

g01atc	Computes univariate summary information: mean, variance, skewness, kurtosis
g01auc	Combines multiple sets of summary information, for use after nag_summary_stats_onevar (g01atc)
g01hdc	Computes the probability for the multivariate Student's t-distribution
g01lbc	Computes a vector of values for the probability density function of the multivariate Normal distribution
g01wac	Computes the mean and standard deviation using a rolling window
g02ajc	Computes the nearest correlation matrix to a real square matrix, using element-wise weighting
g02bzc	Combines two sums of squares matrices, for use after nag_sum_sqs (g02buc)
g03gac	Fits a Gaussian mixture model
g05xac	Initializes the Brownian bridge generator
g05xbc	Generate paths for a free or non-free Wiener process using the Brownian bridge algorithm
g05xcc	Initializes the generator which backs out the increments of sample paths generated by a Brownian bridge algorithm
g05xdc	Backs out the increments from sample paths generated by a Brownian bridge algorithm
g05xec	Creates a Brownian bridge construction order out of a set of input times
g05zmc	Setup for simulating one-dimensional random fields, user-defined variogram
g05znc	Setup for simulating one-dimensional random fields
g05zpc	Generates realizations of a one-dimensional random field
g05zqc	Setup for simulating two-dimensional random fields, user-defined variogram
g05zrc	Setup for simulating two-dimensional random fields, preset variogram
g05zsc	Generates realizations of a two-dimensional random field
g05ztc	Generates realizations of fractional Brownian motion
g10bbc	Kernel density estimate using Gaussian kernel (thread safe)
g13mec	Computes the iterated exponential moving average for a univariate inhomogeneous time series
g13mfc	Computes the iterated exponential moving average for a univariate inhomogeneous time series, intermediate results are also returned
g13mgc	Computes the exponential moving average for a univariate inhomogeneous time series
h05aac	Best n subsets of size p (reverse communication)
h05abc	Best n subsets of size p (direct communication)
s22bac	Real confluent hypergeometric function ${}_1F_1(a;b;x)$
s22bbc	Real confluent hypergeometric function ${}_1F_1(a;b;x)$ in scaled form
s22bec	Real Gauss hypergeometric function ${}_2F_1(a,b;c;x)$
s22bfc	Real Gauss hypergeometric function ${}_2F_1(a,b;c;x)$ in scaled form.
s30ncc	Heston's model option pricing with term structure
x07aac	Determines whether its argument has a finite value
x07abc	Determines whether its argument is a NaN (Not A Number)
x07bac	Creates a signed infinite value.

x07bbc	Creates a NaN (Not A Number)
x07cac	Gets current behaviour of floating-point exceptions
x07cbc	Sets behaviour of floating-point exceptions

3 Withdrawn Functions

The following functions have been withdrawn from the NAG C Library at Mark 24. Warning of their withdrawal was included in the NAG C Library Manual at Mark 23, together with advice on which functions to use instead. See the document 'Advice on Replacement Calls for Withdrawn/Superseded Functions' for more detailed guidance.

Withdrawn Function	**Replacement Function(s)**
c05adc	nag_zero_cont_func_brent (c05ayc)
c05nbc	nag_zero_nonlin_eqns_easy (c05qbc)
c05pbc	nag_zero_nonlin_eqns_deriv_easy (c05rbc)
c05tbc	nag_zero_nonlin_eqns_easy (c05qbc)
c05zbc	nag_check_derivs (c05zdc)
c05zcc	nag_check_derivs (c05zdc)
d01ajc	nag_1d_quad_gen_1 (d01sjc)
d01akc	nag_1d_quad_osc_1 (d01skc)
d01alc	nag_1d_quad_brkpts_1 (d01slc)
d01amc	nag_1d_quad_inf_1 (d01smc)
d01anc	nag_1d_quad_wt_trig_1 (d01snc)
d01apc	nag_1d_quad_wt_alglog_1 (d01spc)
d01aqc	nag_1d_quad_wt_cauchy_1 (d01sqc)
d01asc	nag_1d_quad_inf_wt_trig_1 (d01ssc)
d01bac	nag_quad_1d_gauss_vec (d01uac)
e04ccc	nag_opt_simplex_easy (e04cbc)
g01cec	nag_deviates_normal (g01fac)
g05cac	nag_rand_basic (g05sac)
g05cbc	nag_rand_init_repeatable (g05kfc)
g05ccc	nag_rand_init_nonrepeatable (g05kgc)
g05cfc	No longer required.
g05cgc	No longer required.
g05dac	nag_rand_uniform (g05sqc)
g05dbc	nag_rand_exp (g05sfc)
g05ddc	nag_rand_normal (g05skc)
g05dyc	nag_rand_discrete_uniform (g05tlc)
g05eac	nag_rand_matrix_multi_normal (g05rzc)
g05ecc	nag_rand_poisson (g05tjc)
g05edc	nag_rand_binomial (g05tac)
g05ehc	nag_rand_permute (g05ncc)

g05ejc	nag_rand_sample (g05ndc)
g05exc	nag_rand_gen_discrete (g05tdc)
g05eyc	nag_rand_gen_discrete (g05tdc)
g05ezc	nag_rand_matrix_multi_normal (g05rzc)
g05fec	nag_rand_beta (g05sbc)
g05ffc	nag_rand_gamma (g05sjc)
g05hac	nag_rand_arma (g05phc)
g05hkc	nag_rand_agarchI (g05pdc)
g05hlc	nag_rand_agarchII (g05pec)
g05hmc	nag_rand_garchGJR (g05pfc)
g05kac	nag_rand_basic (g05sac)
g05kbc	nag_rand_init_repeatable (g05kfc)
g05kcc	nag_rand_init_nonrepeatable (g05kgc)
g05kec	nag_rand_logical (g05tbc)
g05lac	nag_rand_normal (g05skc)
g05lbc	nag_rand_students_t (g05snc)
g05lcc	nag_rand_chi_sq (g05sdc)
g05ldc	nag_rand_f (g05shc)
g05lec	nag_rand_beta (g05sbc)
g05lfc	nag_rand_gamma (g05sjc)
g05lgc	nag_rand_uniform (g05sqc)
g05lhc	nag_rand_triangular (g05spc)
g05ljc	nag_rand_exp (g05sfc)
g05lkc	nag_rand_lognormal (g05smc)
g05llc	nag_rand_cauchy (g05scc)
g05lmc	nag_rand_weibull (g05ssc)
g05lnc	nag_rand_logistic (g05slc)
g05lpc	nag_rand_von_mises (g05src)
g05lqc	nag_rand_exp_mix (g05sgc)
g05lxc	nag_rand_matrix_multi_students_t (g05ryc)
g05lyc	nag_rand_matrix_multi_normal (g05rzc)
g05lzc	nag_rand_matrix_multi_normal (g05rzc)
g05mac	nag_rand_discrete_uniform (g05tlc)
g05mbc	nag_rand_geom (g05tcc)
g05mcc	nag_rand_neg_bin (g05thc)
g05mdc	nag_rand_logarithmic (g05tfc)
g05mec	nag_rand_compd_poisson (g05tkc)
g05mjc	nag_rand_binomial (g05tac)
g05mkc	nag_rand_poisson (g05tjc)

g05mlc	nag_rand_hypergeometric (g05tec)
g05mrc	nag_rand_gen_multinomial (g05tgc)
g05mzc	nag_rand_gen_discrete (g05tdc)
g05nac	nag_rand_permute (g05ncc)
g05nbc	nag_rand_sample (g05ndc)
g05pac	nag_rand_arma (g05phc)
g05pcc	nag_rand_varma (g05pjc)
g05qac	nag_rand_orthog_matrix (g05pxc)
g05qbc	nag_rand_corr_matrix (g05pyc)
g05qdc	nag_rand_2_way_table (g05pzc)
g05rac	nag_rand_copula_normal (g05rdc)
g05rbc	nag_rand_copula_students_t (g05rcc)
g05yac	nag_quasi_init (g05ylc) and nag_quasi_rand_uniform (g05ymc)
g05ybc	nag_quasi_rand_normal (g05yjc) and nag_quasi_init (g05ylc)
x02dac	No longer required.
x02djc	No longer required.

4 Functions Scheduled for Withdrawal

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

The following functions will be withdrawn at Mark 25.

Functions Scheduled for Withdrawal	**Replacement Function(s)**
c05agc	nag_zero_cont_func_brent_binsrch (c05auc)
c05sdc	nag_zero_cont_func_brent (c05ayc)
c05ubc	nag_zero_nonlin_eqns_deriv_easy (c05rbc)
d01fcc	nag_multid_quad_adapt_1 (d01wcc)
d01gbc	nag_multid_quad_monte_carlo_1 (d01xbc)
f01bnc	nag_zpotrf (f07frc)
f01qcc	nag_dgeqrf (f08aec)
f01qdc	nag_dormqr (f08agc)
f01qec	nag_dorgqr (f08afc)
f01rcc	nag_zgeqrf (f08asc)
f01rdc	nag_zunmqr (f08auc)
f01rec	nag_zungqr (f08atc)
f03aec	nag_dpotrf (f07fdc) and nag_det_real_sym (f03bfc)
f03afc	nag_dgetrf (f07adc) and nag_det_real_gen (f03bac)

f03ahc	nag_zgetrf (f07arc) and nag_det_complex_gen (f03bnc)
f04adc	nag_complex_gen_lin_solve (f04cac)
f04agc	nag_dpotrs (f07fec)
f04ajc	nag_dgetrs (f07aec)
f04akc	nag_zgetrs (f07asc)
f04arc	nag_real_gen_lin_solve (f04bac)
f04awc	nag_zpotrs (f07fsc)
g02ewc	nag_full_step_regsn_monfun (g02efh)
x04aec	No replacement required.

The following functions have been superseded, but will not be withdrawn from the Library until Mark 26 at the earliest.

Superseded Function	**Replacement Function(s)**
c06eac	nag_sum_fft_realherm_1d (c06pac)
c06ebc	nag_sum_fft_realherm_1d (c06pac)
c06ecc	nag_sum_fft_complex_1d (c06pcc)
c06ekc	nag_sum_convcorr_real (c06fkc)
c06frc	nag_sum_fft_complex_1d_multi (c06psc)
c06fuc	nag_sum_fft_complex_2d (c06puc)
c06gbc	No replacement required
c06gcc	No replacement required
c06hac	nag_sum_fft_sine (c06rec)
c06hbc	nag_sum_fft_cosine (c06rfc)
c06hcc	nag_sum_fft_qtrsine (c06rgc)
c06hdc	nag_sum_fft_qtrcosine (c06rhc)
d02pcc	nag_ode_ivp_rkts_range (d02pec) and associated d02p functions
d02pdc	nag_ode_ivp_rkts_onestep (d02pfc) and associated d02p functions
d02ppc	No replacement required
d02pvc	nag_ode_ivp_rkts_setup (d02pqc)
d02pwc	nag_ode_ivp_rkts_reset_tend (d02prc)
d02pxc	nag_ode_ivp_rkts_interp (d02psc)
d02pzc	nag_ode_ivp_rkts_errass (d02puc)
e04jbc	nag_opt_nlp (e04ucc)
f02aac	nag_dsyev (f08fac)
f02abc	nag_dsyev (f08fac)
f02adc	nag_dsygv (f08sac)
f02aec	nag_dsygv (f08sac)
f02afc	nag_dgeev (f08nac)
f02agc	nag_dgeev (f08nac)

f02awc	nag_zheev (f08fnc)
f02axc	nag_zheev (f08fnc)
f02bjc	nag_dggev (f08wac)
f02wec	nag_dgesvd (f08kbc)
f02xec	nag_zgesvd (f08kpc)
g01aac	nag_summary_stats_onevar (g01atc)
g10bac	nag_kernel_density_gauss (g10bbc)