GAMS Index for the NAG C Library

This index classifies NAG C Library routines according to Version 2 of the GAMS classification scheme described in [1]. Note that only those GAMS classes which contain Library routines, either directly or in a subclass, are included below.

C Elementary and special functions (*search also class L5)*
C3 Polynomials
C3a Orthogonal
C3a2 Chebyshev, Legendre
`e02aec` nag_1d_cheb_eval
Evaluates the coefficients of a Chebyshev series polynomial
C4 Elementary transcendental functions
C4c Hyperbolic, inverse hyperbolic
`s10aac` nag_tanh
Hyperbolic tangent, tanh x
`s10abc` nag_sinh
Hyperbolic sine, sinh x
`s10acc` nag_cosh
Hyperbolic cosine, cosh x
`s11aac` nag_arctanh
Inverse hyperbolic tangent, arctanh x
`s11abc` nag_arcsinh
Inverse hyperbolic sine, arcsinh x
`s11acc` nag_arccosh
Inverse hyperbolic cosine, arccosh x
C5 Exponential and logarithmic integrals
`s13aac` nag_exp_integral
Exponential integral $E_1(x)$
C6 Cosine and sine integrals
`s13acc` nag_cos_integral
Cosine integral $\mathrm{Ci}(x)$
`s13adc` nag_sin_integral
Sine integral $\mathrm{Si}(x)$
C7 Gamma
C7a Gamma, log gamma, reciprocal gamma
`s14aac` nag_gamma
Gamma function $\Gamma(x)$
`s14abc` nag_log_gamma
Log Gamma function $\ln(\Gamma(x))$
C7e Incomplete gamma
`s14bac` nag_incomplete_gamma
Incomplete gamma functions $P(a,x)$ and $Q(a,x)$
C8 Error functions
C8a Error functions, their inverses, integrals, including the normal distribution function
`s15abc` nag_cumul_normal
Cumulative normal distribution function, $P(x)$
`s15acc` nag_cumul_normal_complem
Complement of cumulative normal distribution function, $Q(x)$
`s15adc` nag_erfc
Complement of error function, erfc x
`s15aec` nag_erf
Error function, erf x
C8b Fresnel integrals
`s20acc` nag_fresnel_s
Fresnel integral $S(x)$
`s20adc` nag_fresnel_c
Fresnel integral $C(x)$
C10 Bessel functions
C10a J, Y, H_1, H_2
C10a1 Real argument, integer order
`s17acc` nag_bessel_y0
Bessel function $Y_0(x)$
`s17adc` nag_bessel_y1
Bessel function $Y_1(x)$
`s17aec` nag_bessel_j0
Bessel function $J_0(x)$

H2a1a Integrand available via user-defined procedure
H2a1a1 Automatic (user need only specify required accuracy)
`d01ajc` nag_1d_quad_gen
1-D adaptive quadrature, allowing for badly-behaved integrands
`d01akc` nag_1d_quad_osc
1-D adaptive quadrature, suitable for oscillating functions
`d01alc` nag_1d_quad_brkpts
1-D adaptive quadrature, allowing for singularities at specified points
`d01sjc` nag_1d_quad_gen_1
1-D adaptive quadrature, allowing for badly-behaved integrands, thread-safe
`d01skc` nag_1d_quad_osc_1
1-D adaptive quadrature, suitable for oscillating functions, thread-safe
`d01slc` nag_1d_quad_brkpts_1
1-D adaptive quadrature, allowing for singularities at specified points, thread-safe
H2a1a2 Nonautomatic
`d01bac` nag_1d_quad_gauss
1-D Gaussian quadrature rule evaluation
`d01tac` nag_1d_quad_gauss_1
1-D Gaussian quadrature rule evaluation, thread-safe
H2a1b Integrand available only on grid
H2a1b2 Nonautomatic
`d01gac` nag_1d_quad_vals
1-D integration of a function defined by data values only
H2a2 Finite interval (specific or special type integrand including weight functions, oscillating and singular integrands, principal value integrals, splines, etc.)
H2a2a Integrand available via user-defined procedure
H2a2a1 Automatic (user need only specify required accuracy)
`d01anc` nag_1d_quad_wt_trig
1-D adaptive quadrature, finite interval, sine or cosine weight functions
`d01apc` nag_1d_quad_wt_alglog
1-D adaptive quadrature, weight function with end-point singularities of algebraic-logarithmic type
`d01aqc` nag_1d_quad_wt_cauchy
1-D adaptive quadrature, weight function $1/(x-c)$, Cauchy principal value
`d01skc` nag_1d_quad_osc_1
1-D adaptive quadrature, suitable for oscillating functions, thread-safe
`d01slc` nag_1d_quad_brkpts_1
1-D adaptive quadrature, allowing for singularities at specified points, thread-safe
`d01snc` nag_1d_quad_wt_trig_1
1-D adaptive quadrature, finite interval, sine or cosine weight functions, thread-safe
`d01spc` nag_1d_quad_wt_alglog_1
1-D adaptive quadrature, weight function with end-point singularities of algebraic-logarithmic type, thread-safe
`d01sqc` nag_1d_quad_wt_cauchy_1
1-D adaptive quadrature, weight function $1/(x-c)$, Cauchy principal value, thread-safe
H2a2b Integrand available only on grid
H2a2b1 Automatic (user need only specify required accuracy)
`e02bdc` nag_1d_spline_intg
Evaluation of fitted cubic spline, definite integral
H2a3 Semi-infinite interval (including e^{-x} weight function)
H2a3a Integrand available via user-defined procedure
H2a3a1 Automatic (user need only specify required accuracy)
`d01amc` nag_1d_quad_inf
1-D adaptive quadrature over infinite or semi-infinite interval
`d01asc` nag_1d_quad_inf_wt_trig
1-D adaptive quadrature, semi-infinite interval, sine or cosine weight function
`d01smc` nag_1d_quad_inf_1
1-D adaptive quadrature over infinite or semi-infinite interval, thread-safe
`d01ssc` nag_1d_quad_inf_wt_trig_1
1-D adaptive quadrature, semi-infinite interval, sine or cosine weight function, thread-safe
H2a3a2 Nonautomatic
`d01bac` nag_1d_quad_gauss
1-D Gaussian quadrature rule evaluation
`d01tac` nag_1d_quad_gauss_1
1-D Gaussian quadrature rule evaluation, thread-safe
H2a4 Infinite interval (including e^{-x^2} weight function)
H2a4a Integrand available via user-defined procedure

L10 Time series analysis (*search also class J)*
L10a Univariate (*search also classes L3a6 and L3a7)*
L10a2 Time domain analysis
L10a2a Summary statistics
L10a2a1 Autocorrelations and autocovariances
`g13abc` nag_tsa_auto_corr
Sample autocorrelation function
L10a2a2 Partial autocorrelations
`g13acc` nag_tsa_auto_corr_part
Partial autocorrelation function
L10a2c Autoregressive models
L10a2c1 Model identification
`g13acc` nag_tsa_auto_corr_part
Partial autocorrelation function
L10a2d ARMA and ARIMA models (including Box–Jenkins methods)
L10a2d2 Parameter estimation
`g13bec` nag_tsa_multi_inp_model_estim
Estimation for time series models
L10a2e State-space analysis (e.g., Kalman filtering)
`g13eac` nag_kalman_sqrt_filt_cov_var
One iteration step of the time-varying Kalman filter recursion using the square root covariance implementation
`g13ebc` nag_kalman_sqrt_filt_cov_invar
One iteration step of the time-invariant Kalman filter recursion using the square root covariance implementation with (A, C) in lower observer Hessenberg form
`g13ecc` nag_kalman_sqrt_filt_info_var
One iteration step of the time-varying Kalman filter recursion using the square root information implementation
`g13edc` nag_kalman_sqrt_filt_info_invar
One iteration step of the time-invariant Kalman filter recursion using the square root information implementation with $(A^{-1}, A^{-1}B)$ in upper controller Hessenberg form
`g13ewc` nag_trans_hessenberg_observer
Unitary state-space transformation to reduce (A, C) to lower or upper observer Hessenberg form
`g13exc` nag_trans_hessenberg_controller
Unitary state-space transformation to reduce (B, A) to lower or upper controller Hessenberg form
L10a3 Frequency domain analysis (*search also class J1)*
L10a3a Spectral analysis
L10a3a3 Spectrum estimation using the periodogram
`g13cbc` nag_tsa_spectrum_univar
Univariate time series, smoothed sample spectrum using spectral smoothing by the trapezium frequency (Daniell) window
L10b Two time series (*search also classes L3b3c, L10c, and L10d)*
L10b2 Time domain analysis
L10b2b Transfer function models
`g13bec` nag_tsa_multi_inp_model_estim
Estimation for time series models
`g13bjc` nag_tsa_multi_inp_model_forecast
Forecasting function
L10b3 Frequency domain analysis (*search also class J1)*
L10b3a Cross-spectral analysis
L10b3a3 Cross-spectrum estimation using the cross-periodogram
`g13cdc` nag_tsa_spectrum_bivar
Multivariate time series, smoothed sample cross spectrum using spectral smoothing by the trapezium frequency (Daniell) window
L10b3a6 Spectral functions
`g13cec` nag_tsa_cross_spectrum_bivar
Multivariate time series, cross amplitude spectrum, squared coherency, bounds, univariate and bivariate (cross) spectra
`g13cfc` nag_tsa_gain_phase_bivar
Multivariate time series, gain, phase, bounds, univariate and bivariate (cross) spectra
`g13cgc` nag_tsa_noise_spectrum_bivar
Multivariate time series, noise spectrum, bounds, impulse response function and its standard error
L12 Discriminant analysis
`g03acc` nag_mv_canon_var
Canonical variate analysis

L4a1a4 Parameter estimates and tests
L4a1a4n Normal
`g01ddc` nag_shapiro_wilk_test
Shapiro and Wilk's W test for Normality
`g07cac` nag_2_sample_t_test
t-test statistic, for a difference in means between two Normal populations, confidence interval
L4a5 Categorical data
`g11aac` nag_chi_sq_2_way_table
χ^2 statistic for two-way contingency table
L5 Function evaluation (*search also class C)*
L5a Univariate
L5a1 Cumulative distribution functions, probability density functions
L5a1b Beta, binomial
`g01bjc` nag_binomial_dist
Binomial distribution function
`g01eec` nag_prob_beta_dist
Upper and lower tail probabilities and probability density function for the beta distribution
L5a1c Cauchy, χ^2
`g01ecc` nag_prob_chi_sq
Probabilities for χ^2 distribution
L5a1e Error function, exponential, extreme value
`s15adc` nag_erfc
Complement of error function, erfc x
`s15aec` nag_erf
Error function, erf x
L5a1f F distribution
`g01edc` nag_prob_f_dist
Probabilities for F-distribution
L5a1g Gamma, general, geometric
`g01efc` nag_gamma_dist
Probabilities for the gamma distribution
L5a1h Halfnormal, hypergeometric
`g01blc` nag_hypergeom_dist
Hypergeometric distribution function
L5a1n Negative binomial, normal
`g01eac` nag_prob_normal
Probabilities for the standard Normal distribution
`s15abc` nag_cumul_normal
Cumulative normal distribution function, $P(x)$
`s15acc` nag_cumul_normal_complem
Complement of cumulative normal distribution function, $Q(x)$
L5a1p Pareto, Poisson
`g01bkc` nag_poisson_dist
Poisson distribution function
L5a1t t distribution
`g01ebc` nag_prob_students_t
Probabilities for Student's t-distribution
L5a2 Inverse distribution functions, sparsity functions
L5a2b Beta, binomial
`g01fec` nag_deviates_beta
Deviates for the beta distribution
L5a2c Cauchy, χ^2
`g01fcc` nag_deviates_chi_sq
Deviates for the χ^2 distribution
L5a2f F distribution
`g01fdc` nag_deviates_f_dist
Deviates for the F-distribution
L5a2g Gamma, general, geometric
`g01ffc` nag_deviates_gamma_dist
Deviates for the gamma distribution
L5a2n Negative binomial, normal, normal order statistics
`g01cec` nag_deviates_normal_dist
Deviate of Normal distribution function
`g01fac` nag_deviates_normal
Deviates for the Normal distribution

L7b Two-way (*search also class L7d)*
`g04bbc` nag_anova_random
General block design or completely randomized design
L7d Multi-way
L7d1 Balanced complete data (e.g., factorial designs)
`g04cac` nag_anova_factorial
Complete factorial design
L7f Generate experimental designs
`g02dac` nag_regsn_mult_linear
Fits a general (multiple) linear regression model
`g02dnc` nag_regsn_mult_linear_est_func
Estimate of an estimable function for a general linear regression model
L7g Service routines
`g04czc` nag_anova_factorial_free
Complete factorial design
L8 Regression (*search also classes D5, D6, D9, G, K)*
L8a Simple linear (i.e., $y = b_0 + b_1 x$) (*search also class L8h)*
L8a1 Ordinary least squares
L8a1a Parameter estimation
L8a1a1 Unweighted data
`g02cac` nag_simple_linear_regression
Simple linear regression with or without a constant term, data may be weighted
`g02cbc` nag_regress_confid_interval
Simple linear regression confidence intervals for the regression line and individual points
L8b Polynomial (e.g., $y = b_0 + b_1 x + b_2 x^2$) (*search also class L8c)*
L8b1 Ordinary least squares
L8b1b Parameter estimation
L8b1b2 Using orthogonal polynomials
`e02adc` nag_1d_cheb_fit
Computes the coefficients of a Chebyshev series polynomial for arbitrary data
L8c Multiple linear (i.e. $y = b_0 + b_1 x_1 + ... + b_p x_p$)
L8c1 Ordinary least squares
L8c1a Variable selection
L8c1a1 Using raw data
`g02ddc` nag_regsn_mult_linear_upd_model
Estimates of regression parameters from an updated model
`g02dec` nag_regsn_mult_linear_add_var
Add a new independent variable to a general linear regression model
`g02dfc` nag_regsn_mult_linear_delete_var
Delete an independent variable from a general linear regression model
L8c1b Parameter estimation (*search also class L8c1a)*
L8c1b1 Using raw data
`g02dac` nag_regsn_mult_linear
Fits a general (multiple) linear regression model
`g02dcc` nag_regsn_mult_linear_addrem_obs
Add/delete an observation to/from a general linear regression model
`g02ddc` nag_regsn_mult_linear_upd_model
Estimates of regression parameters from an updated model
`g02dec` nag_regsn_mult_linear_add_var
Add a new independent variable to a general linear regression model
`g02dfc` nag_regsn_mult_linear_delete_var
Delete an independent variable from a general linear regression model
`g02dkc` nag_regsn_mult_linear_tran_model
Estimates of parameters of a general linear regression model for given constraints
`g02dnc` nag_regsn_mult_linear_est_func
Estimate of an estimable function for a general linear regression model
L8c1c Analysis (*search also classes L8c1a and L8c1b)*
`g02fac` nag_regsn_std_resid_influence
Calculate standardized residuals and influence statistics
L8c1d Inference (*search also classes L8c1a and L8c1b)*
`g02dnc` nag_regsn_mult_linear_est_func
Estimate of an estimable function for a general linear regression model
L8c2 Several regressions
`g02dgc` nag_regsn_mult_linear_newyvar
Fits a general linear regression model to new dependent variable
L8c4 Robust
`g02hac` nag_robust_m_regsn_estim
Robust regression, standard M-estimates

N6a2b Real

`m01cac` nag_double_sort
Quicksort of set of values of data type double

N8 Permuting

`m01esc` nag_reorder_vector
Reorders set of values of arbitrary data type into the order specified by a set of indices

`m01zac` nag_make_indices
Inverts a permutation converting a rank vector to an index vector or vice versa

R Service routines

`a00aac` nag_implementation_details
Library identification, details of implementation and mark

R1 Machine-dependent constants

`X01AAC` nag_pi
π

`X01ABC` nag_euler_constant
Euler's constant, γ

`X02AHC` nag_max_sine_argument
Largest permissible argument for `sin` and `cos` functions

`X02AJC` nag_machine_precision
Machine precision

`X02AKC` nag_real_smallest_number
Smallest positive model number

`X02ALC` nag_real_largest_number
Largest positive model number

`X02AMC` nag_real_safe_small_number
Safe range of floating-point arithmetic

`X02BEC` nag_decimal_digits
Maximum number of decimal digits that can be represented

`X02BHC` nag_real_base
Parameter b of model of floating-point arithmetic

`X02BJC` nag_real_base_digits
Parameter p of model of floating-point arithmetic

`X02BKC` nag_real_min_exponent
Parameter $e_{\min}$ of model of floating-point arithmetic

`X02BLC` nag_real_max_exponent
Parameter $e_{\max}$ of model of floating-point arithmetic

`X02DAC` nag_underflow_flag
Switch for taking precautions to avoid underflow

`X02DJC` nag_real_arithmetic_rounds
Parameter ROUNDS of model of floating-point arithmetic

References

Boisvert R F, Howe S E and Kahaner D K (1990) The guide to available mathematical software problem classification scheme. *Report NISTIR 4475* Applied and Computational Mathematics Division, National Institute of Standards and Technology.

Boisvert R F, Howe S E and Kahaner D K (1985) GAMS – a framework for the management of scientific software. *ACM Trans. Math. Software* **11** 313-355.

Boisvert R F (1989) The guide to available mathematical software advisory system. *Math. Comput. Simul.* **31** 453-464.