GAMS Index for the NAG Fortran 77 Library

This index classifies NAG Fortran 77 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.

A Arithmetic, error analysis
A3 Real
A3a Standard precision
`F06BLF` Compute quotient of two real scalars, with overflow flag
A4 Complex
A4a Standard precision
`A02ABF` Modulus of a complex number
`A02ACF` Quotient of two complex numbers
`F06CLF` Compute quotient of two complex scalars, with overflow flag
A7 Sequences (e.g., convergence acceleration)
`C06BAF` Acceleration of convergence of sequence, Shanks' transformation and epsilon algorithm
C Elementary and special functions (*search also class L5*)
C1 Integer-valued functions (e.g., factorial, binomial coefficient, permutations, combinations, floor, ceiling)
C10 Bessel functions
C10a J, Y, H_1, H_2
C10a1 Real argument, integer order
`S17ACF` Bessel function $Y_0(x)$
`S17ADF` Bessel function $Y_1(x)$
`S17AEF` Bessel function $J_0(x)$
`S17AFF` Bessel function $J_1(x)$
C10a4 Complex argument, real order
`S17DCF` Bessel functions $Y_{\nu+a}(z)$, real $a \geq 0$, complex z, $\nu = 0, 1, 2, \ldots$
`S17DEF` Bessel functions $J_{\nu+a}(z)$, real $a \geq 0$, complex z, $\nu = 0, 1, 2, \ldots$
`S17DLF` Hankel functions $H^{(j)}_{\nu+a}(z)$, $j = 1, 2$, real $a \geq 0$, complex z, $\nu = 0, 1, 2, \ldots$
C10b I, K
C10b1 Real argument, integer order
`S18ACF` Modified Bessel function $K_0(x)$
`S18ADF` Modified Bessel function $K_1(x)$
`S18AEF` Modified Bessel function $I_0(x)$
`S18AFF` Modified Bessel function $I_1(x)$
`S18CCF` Modified Bessel function $e^x K_0(x)$
`S18CDF` Modified Bessel function $e^x K_1(x)$
`S18CEF` Modified Bessel function $e^{-|x|} I_0(x)$
`S18CFF` Modified Bessel function $e^{-|x|} I_1(x)$
C10b4 Complex argument, real order
`S18DCF` Modified Bessel functions $K_{\nu+a}(z)$, real $a \geq 0$, complex z, $\nu = 0, 1, 2, \ldots$
`S18DEF` Modified Bessel functions $I_{\nu+a}(z)$, real $a \geq 0$, complex z, $\nu = 0, 1, 2, \ldots$
C10c Kelvin functions
`S19AAF` Kelvin function ber x
`S19ABF` Kelvin function bei x
`S19ACF` Kelvin function ker x
`S19ADF` Kelvin function kei x
C10d Airy and Scorer functions
`S17AGF` Airy function $\mathrm{Ai}(x)$
`S17AHF` Airy function $\mathrm{Bi}(x)$
`S17AJF` Airy function $\mathrm{Ai}'(x)$
`S17AKF` Airy function $\mathrm{Bi}'(x)$
`S17DGF` Airy functions $\mathrm{Ai}(z)$ and $\mathrm{Ai}'(z)$, complex z
`S17DHF` Airy functions $\mathrm{Bi}(z)$ and $\mathrm{Bi}'(z)$, complex z
C13 Jacobian elliptic functions, theta functions
`S21CAF` Jacobian elliptic functions sn, cn and dn
C14 Elliptic integrals
`S21BAF` Degenerate symmetrised elliptic integral of 1st kind $R_C(x, y)$
`S21BBF` Symmetrised elliptic integral of 1st kind $R_F(x, y, z)$
`S21BCF` Symmetrised elliptic integral of 2nd kind $R_D(x, y, z)$
`S21BDF` Symmetrised elliptic integral of 3rd kind $R_J(x, y, z, r)$
C2 Powers, roots, reciprocals
`A02AAF` Square root of a complex number
C3 Polynomials
C3a Orthogonal
C3a2 Chebyshev, Legendre
`C06DBF` Sum of a Chebyshev series

D1a2 Minimum and maximum components
- `F06FLF` Elements of real vector with largest and smallest absolute value
- `F06JLF` Index, real vector element with largest absolute value (ISAMAX/IDAMAX)
- `F06JMF` Index, complex vector element with largest absolute value (ICAMAX/IZAMAX)
- `F06KLF` Last non-negligible element of real vector

D1a3 Norm

D1a3a L_1 (sum of magnitudes)
- `F06EKF` Sum the absolute values of real vector elements (SASUM/DASUM)
- `F06JKF` Sum the absolute values of complex vector elements (SCASUM/DZASUM)

D1a3b L_2 (Euclidean norm)
- `F06BMF` Compute Euclidean norm from scaled form
- `F06BNF` Compute square root of $(a^2 + b^2)$, real a and b
- `F06EJF` Compute Euclidean norm of real vector (SNRM2/DNRM2)
- `F06FJF` Update Euclidean norm of real vector in scaled form
- `F06FKF` Compute weighted Euclidean norm of real vector
- `F06JJF` Compute Euclidean norm of complex vector (SCNRM2/DZNRM2)
- `F06KJF` Update Euclidean norm of complex vector in scaled form

D1a3c L_∞ (maximum magnitude)
- `F06FLF` Elements of real vector with largest and smallest absolute value
- `F06JLF` Index, real vector element with largest absolute value (ISAMAX/IDAMAX)
- `F06JMF` Index, complex vector element with largest absolute value (ICAMAX/IZAMAX)

D1a4 Dot product (inner product)
- `F06EAF` Dot product of two real vectors (SDOT/DDOT)
- `F06ERF` Dot product of two real sparse vectors (SDOTI/DDOTI)
- `F06GAF` Dot product of two complex vectors, unconjugated (CDOTU/ZDOTU)
- `F06GBF` Dot product of two complex vectors, conjugated (CDOTC/ZDOTC)
- `F06GRF` Dot product of two complex sparse vector, unconjugated (CDOTUI/ZDOTUI)
- `F06GSF` Dot product of two complex sparse vector, conjugated (CDOTCI/ZDOTCI)
- `X03AAF` Real inner product added to initial value, basic/additional precision
- `X03ABF` Complex inner product added to initial value, basic/additional precision

D1a5 Copy or exchange (swap)
- `F06DFF` Copy integer vector
- `F06EFF` Copy real vector (SCOPY/DCOPY)
- `F06EGF` Swap two real vectors (SSWAP/DSWAP)
- `F06GFF` Copy complex vector (CCOPY/ZCOPY)
- `F06GGF` Swap two complex vectors (CSWAP/ZSWAP)
- `F06KFF` Copy real vector to complex vector

D1a6 Multiplication by scalar
- `F06EDF` Multiply real vector by scalar (SSCAL/DSCAL)
- `F06FDF` Multiply real vector by scalar, preserving input vector
- `F06FGF` Negate real vector
- `F06GDF` Multiply complex vector by complex scalar (CSCAL/ZSCAL)
- `F06HDF` Multiply complex vector by complex scalar, preserving input vector
- `F06HGF` Negate complex vector
- `F06JDF` Multiply complex vector by real scalar (CSSCAL/ZDSCAL)
- `F06KDF` Multiply complex vector by real scalar, preserving input vector

D1a7 Triad ($\alpha x + y$ for vectors x, y and scalar α)
- `F06ECF` Add scalar times real vector to real vector (SAXPY/DAXPY)
- `F06ETF` Add a scalar times a real sparse vector to another real sparse vector (SAXPYI/DAXPYI)
- `F06GCF` Add scalar times complex vector to complex vector (CAXPY/ZAXPY)
- `F06GTF` Add a scalar times a complex sparse vector to another complex sparse vector (CAXPYI/ZAXPYI)

D1a8 Elementary rotation (Givens transformation)
- `F06AAF` Generate real plane rotation (SROTG/DROTG)
- `F06BAF` Generate real plane rotation, storing tangent
- `F06BEF` Generate real Jacobi plane rotation
- `F06BHF` Apply real similarity rotation to 2 by 2 symmetric matrix
- `F06CAF` Generate complex plane rotation, storing tangent, real cosine
- `F06CBF` Generate complex plane rotation, storing tangent, real sine
- `F06CHF` Apply complex similarity rotation to 2 by 2 Hermitian matrix
- `F06EPF` Apply real plane rotation (SROT/DROT)
- `F06EXF` Apply plane rotation to two real sparse vectors (SROTI/DROTI)
- `F06FPF` Apply real symmetric plane rotation to two vectors
- `F06FQF` Generate sequence of real plane rotations
- `F06HPF` Apply complex plane rotation
- `F06HQF` Generate sequence of complex plane rotations
- `F06KPF` Apply real plane rotation to two complex vectors

D1a9 Elementary reflection (Householder transformation)
- `F06FRF` Generate real elementary reflection, NAG style
- `F06FSF` Generate real elementary reflection, LINPACK style

I1a Initial value problems
I1a1 General, nonstiff or mildly stiff
I1a1a One-step methods (e.g., Runge–Kutta)

- D02BGF ODEs, IVP, Runge–Kutta–Merson method, until a component attains given value (simple driver)
- D02BHF ODEs, IVP, Runge–Kutta–Merson method, until function of solution is zero (simple driver)
- D02BJF ODEs, IVP, Runge–Kutta method, until function of solution is zero, integration over range with intermediate output (simple driver)
- D02LAF 2nd order ODEs, IVP, Runge–Kutta–Nystrom method
- D02PCF ODEs, IVP, Runge–Kutta method, integration over range with output
- D02PDF ODEs, IVP, Runge–Kutta method, integration over one step

I1a1b Multistep methods (e.g. Adams predictor-corrector)

- D02CJF ODEs, IVP, Adams method, until function of solution is zero, intermediate output (simple driver)
- D02QFF ODEs, IVP, Adams method with root-finding (forward communication, comprehensive)
- D02QGF ODEs, IVP, Adams method with root-finding (reverse communication, comprehensive)

I1a2 Stiff and mixed algebraic- differential equations

- D02EJF ODEs, stiff IVP, BDF method, until function of solution is zero, intermediate output (simple driver)
- D02NBF Explicit ODEs, stiff IVP, full Jacobian (comprehensive)
- D02NCF Explicit ODEs, stiff IVP, banded Jacobian (comprehensive)
- D02NDF Explicit ODEs, stiff IVP, sparse Jacobian (comprehensive)
- D02NGF Implicit/algebraic ODEs, stiff IVP, full Jacobian (comprehensive)
- D02NHF Implicit/algebraic ODEs, stiff IVP, banded Jacobian (comprehensive)
- D02NJF Implicit/algebraic ODEs, stiff IVP, sparse Jacobian (comprehensive)
- D02NMF Explicit ODEs, stiff IVP (reverse communication, comprehensive)
- D02NNF Implicit/algebraic ODEs, stiff IVP (reverse communication, comprehensive)
- D03PKF General system of first order PDEs, coupled DAEs, method of lines, Keller box discretisation, one space variable
- D03PPF General system of parabolic PDEs, coupled DAEs, method of lines, finite differences, remeshing, one space variable
- D03PRF General system of first order PDEs, coupled DAEs, method of lines, Keller box discretisation, remeshing, one space variable

I1b Multipoint boundary value problems
I1b1 Linear

- D02GBF ODEs, boundary value problem, finite difference technique with deferred correction, general linear problem
- D02JAF ODEs, boundary value problem, collocation and least-squares, single nth order linear equation
- D02JBF ODEs, boundary value problem, collocation and least-squares, system of 1st order linear equations
- D02TGF nth order linear ODEs, boundary value problem, collocation and least-squares

I1b2 Nonlinear

- D02AGF ODEs, boundary value problem, shooting and matching technique, allowing interior matching point, general parameters to be determined
- D02GAF ODEs, boundary value problem, finite difference technique with deferred correction, simple nonlinear problem
- D02HAF ODEs, boundary value problem, shooting and matching, boundary values to be determined
- D02HBF ODEs, boundary value problem, shooting and matching, general parameters to be determined
- D02RAF ODEs, general nonlinear boundary value problem, finite difference technique with deferred correction, continuation facility
- D02SAF ODEs, boundary value problem, shooting and matching technique, subject to extra algebraic equations, general parameters to be determined
- D02TKF ODEs, general nonlinear boundary value problem, collocation technique

I1b3 Eigenvalue (e.g., Sturm-Liouville)

- D02AGF ODEs, boundary value problem, shooting and matching technique, allowing interior matching point, general parameters to be determined
- D02HBF ODEs, boundary value problem, shooting and matching, general parameters to be determined
- D02KAF 2nd order Sturm–Liouville problem, regular system, finite range, eigenvalue only
- D02KDF 2nd order Sturm–Liouville problem, regular/singular system, finite/infinite range, eigenvalue only, user-specified break-points
- D02KEF 2nd order Sturm–Liouville problem, regular/singular system, finite/infinite range, eigenvalue and eigenfunction, user-specified break-points

I1c Service routines (e.g., interpolation of solutions, error handling, test programs)

- D02LXF 2nd order ODEs, IVP, set-up for D02LAF
- D02LYF 2nd order ODEs, IVP, diagnostics for D02LAF

References

[1] 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.

[2] 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.

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