NAG Toolbox

nag_roots_lambertw_real (c05ba)

1 Purpose

nag_roots_lambertw_real (c05ba) returns the real values of Lambert's W function $W(x)$.

2 Syntax

```
[result, ifail] = nag_roots_lambertw_real(x, branch, offset)
[result, ifail] = c05ba(x, branch, offset)
```

3 Description

nag_roots_lambertw_real (c05ba) calculates an approximate value for the real branches of Lambert's W function (sometimes known as the 'product log' or 'Omega' function), which is the inverse function of

$$f(w) = we^w \quad \text{for} \quad w \in C.$$

The function f is many-to-one, and so, except at 0, W is multivalued. nag_roots_lambertw_real (c05ba) restricts W and its argument x to be real, resulting in a function defined for $x \geq -\exp(-1)$ and which is double valued on the interval $(-\exp(-1), 0)$. This double-valued function is split into two real-valued branches according to the sign of $W(x) + 1$. We denote by W_0 the branch satisfying $W_0(x) \geq -1$ for all real x, and by W_{-1} the branch satisfying $W_{-1}(x) \leq -1$ for all real x. You may select your branch of interest using the argument **branch**.

The precise method used to approximate W is described fully in Barry *et al.* (1995). For x close to $-\exp(-1)$ greater accuracy comes from evaluating $W(-\exp(-1) + \Delta x)$ rather than $W(x)$: by setting **offset** $= \mathit{true}$ on entry you inform nag_roots_lambertw_real (c05ba) that you are providing Δx, not x, in **x**.

4 References

Barry D J, Culligan–Hensley P J, and Barry S J (1995) Real values of the W-function *ACM Trans. Math. Software* **21(2)** 161–171

5 Parameters

5.1 Compulsory Input Parameters

1: **x** – REAL (KIND=nag_wp)

If **offset** $= \mathit{true}$, **x** is the offset Δx from $-\exp(-1)$ of the intended argument to W; that is, $W(\beta)$ is computed, where $\beta = -\exp(-1) + \Delta x$.

If **offset** $= \mathit{false}$, **x** is the argument x of the function; that is, $W(\beta)$ is computed, where $\beta = x$.

Constraints:

if **branch** $= 0$, $-\exp(-1) \leq \beta$;
if **branch** $= -1$, $-\exp(-1) \leq \beta < 0.0$.

2: **branch** – INTEGER

The real branch required.

branch $= 0$
The branch W_0 is selected.

branch $= -1$
The branch W_{-1} is selected.

Constraint: **branch** $= 0$ or -1.

3: **offset** – LOGICAL

Controls whether or not **x** is being specified as an offset from $-\exp(-1)$.

5.2 Optional Input Parameters

None.

5.3 Output Parameters

1: **result**

The result of the function.

2: **ifail** – INTEGER

ifail $= 0$ unless the function detects an error (see Section 5).

6 Error Indicators and Warnings

Note: nag_roots_lambertw_real (c05ba) may return useful information for one or more of the following detected errors or warnings.

Errors or warnings detected by the function:

ifail $= 1$

Constraint: **branch** $= 0$ or -1.

Constraint: if **branch** $= -1$ and **offset** $= \mathit{false}$ then $\mathbf{x} < 0.0$.

Constraint: if **branch** $= -1$ and **offset** $= \mathit{true}$ then $\mathbf{x} < \exp(-1.0)$.

Constraint: if **offset** $= \mathit{false}$ then $\mathbf{x} \geq -\exp(-1.0)$.

Constraint: if **offset** $= \mathit{true}$ then $\mathbf{x} \geq 0.0$.

ifail $= 2$ (*warning*)

For the given offset **x**, W is negligibly different from -1.

x is close to $-\exp(-1)$.

ifail $= -99$

An unexpected error has been triggered by this routine. Please contact NAG.

ifail $= -399$

Your licence key may have expired or may not have been installed correctly.

ifail $= -999$

Dynamic memory allocation failed.

7 Accuracy

For a high percentage of legal **x** on input, nag_roots_lambertw_real (c05ba) is accurate to the number of decimal digits of precision on the host machine (see nag_machine_decimal_digits (x02be)). An extra digit may be lost on some implementations and for a small proportion of such **x**. This depends on the accuracy of the base-10 logarithm on your system.

8 Further Comments

None.

9 Example

This example reads from a file the values of the required branch, whether or not the arguments to W are to be considered as offsets to $-\exp(-1)$, and the arguments **x** themselves. It then evaluates the function for these sets of input data **x** and prints the results.

9.1 Program Text

```
     function c05ba_example

fprintf('c05ba example results\n\n');

branch = nag_int(0);
offset = false;
x = [0.5, 1.0, 4.5, 6.0, 7.0e7];
w = zeros(length(x),1);
ifails = zeros(length(x),1);
for i = 1:length(x)
  [w(i), ifails(i)] = c05ba(x(i), branch, offset);
end
fprintf('\nBranch = %d\n', branch);
if offset
  fprintf('Offset = true\n\n');
else
  fprintf('Offset = false\n\n');
end
fprintf('      x            w(x)      ifail\n');
for i=1:5
  fprintf('%13.5e %13.5e    %d\n', x(i), w(i), ifails(i));
end
```

9.2 Program Results

```
     c05ba example results

Branch = 0
Offset = false

      x            w(x)      ifail
  5.00000e-01   3.51734e-01    0
  1.00000e+00   5.67143e-01    0
  4.50000e+00   1.26724e+00    0
  6.00000e+00   1.43240e+00    0
  7.00000e+07   1.53339e+01    0
```