NAG Toolbox

nag_specfun_kelvin_ker (s19ac)

1 Purpose

nag_specfun_kelvin_ker (s19ac) returns a value for the Kelvin function $\operatorname{ker} x$, via the function name.

2 Syntax

```
[result, ifail] = nag_specfun_kelvin_ker(x)
[result, ifail] = s19ac(x)
```

3 Description

nag_specfun_kelvin_ker (s19ac) evaluates an approximation to the Kelvin function $\operatorname{ker} x$.

Note: for $x < 0$ the function is undefined and at $x = 0$ it is infinite so we need only consider $x > 0$.

The function is based on several Chebyshev expansions:

For $0 < x \leq 1$,

$$\operatorname{ker} x = -f(t)\log(x) + \frac{\pi}{16}x^2 g(t) + y(t)$$

where $f(t)$, $g(t)$ and $y(t)$ are expansions in the variable $t = 2x^4 - 1$.

For $1 < x \leq 3$,

$$\operatorname{ker} x = \exp\left(-\frac{11}{16}x\right)q(t)$$

where $q(t)$ is an expansion in the variable $t = x - 2$.

For $x > 3$,

$$\operatorname{ker} x = \sqrt{\frac{\pi}{2x}}e^{-x/\sqrt{2}}\left[\left(1 + \frac{1}{x}c(t)\right)\cos\beta - \frac{1}{x}d(t)\sin\beta\right]$$

where $\beta = \frac{x}{\sqrt{2}} + \frac{\pi}{8}$, and $c(t)$ and $d(t)$ are expansions in the variable $t = \frac{6}{x} - 1$.

When x is sufficiently close to zero, the result is computed as

$$\operatorname{ker} x = -\gamma - \log\left(\frac{x}{2}\right) + \left(\pi - \frac{3}{8}x^2\right)\frac{x^2}{16}$$

and when x is even closer to zero, simply as $\operatorname{ker} x = -\gamma - \log\left(\frac{x}{2}\right)$.

For large x, $\operatorname{ker} x$ is asymptotically given by $\sqrt{\frac{\pi}{2x}}e^{-x/\sqrt{2}}$ and this becomes so small that it cannot be computed without underflow and the function fails.

4 References

Abramowitz M and Stegun I A (1972) *Handbook of Mathematical Functions* (3rd Edition) Dover Publications

5 Parameters

5.1 Compulsory Input Parameters

1: **x** – REAL (KIND=nag_wp)

The argument x of the function.

Constraint: **x** > 0.0.

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

Errors or warnings detected by the function:

ifail $= 1$

On entry, **x** is too large: the result underflows. On softfailure, the function returns zero.

ifail $= 2$

On entry, **x** ≤ 0.0: the function is undefined. On softfailure the function returns zero.

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

Let E be the absolute error in the result, ϵ be the relative error in the result and δ be the relative error in the argument. If δ is somewhat larger than the ***machine precision***, then we have:

$$E \simeq \left| \frac{x}{\sqrt{2}} (\mathrm{ker}_1\, x + \mathrm{kei}_1\, x) \right| \delta,$$

$$\epsilon \simeq \left| \frac{x}{\sqrt{2}} \frac{\mathrm{ker}_1\, x + \mathrm{kei}_1\, x}{\mathrm{ker}\, x} \right| \delta.$$

For very small x, the relative error amplification factor is approximately given by $\frac{1}{|\log(x)|}$, which implies a strong attenuation of relative error. However, ϵ in general cannot be less than the ***machine precision***.

For small x, errors are damped by the function and hence are limited by the ***machine precision***.

For medium and large x, the error behaviour, like the function itself, is oscillatory, and hence only the absolute accuracy for the function can be maintained. For this range of x, the amplitude of the absolute error decays like $\sqrt{\frac{\pi x}{2}} e^{-x/\sqrt{2}}$ which implies a strong attenuation of error. Eventually, $\ker x$, which asymptotically behaves like $\sqrt{\frac{\pi}{2x}} e^{-x/\sqrt{2}}$, becomes so small that it cannot be calculated without causing underflow, and the function returns zero. Note that for large x the errors are dominated by those of the standard function exp.

8 Further Comments

Underflow may occur for a few values of x close to the zeros of $\ker x$, below the limit which causes a failure with **ifail** $= 1$.

9 Example

This example reads values of the argument x from a file, evaluates the function at each value of x and prints the results.

9.1 Program Text

```
function s19ac_example

fprintf('s19ac example results\n\n');

x = [0.1   1    2.5   5   10   15];
n = size(x,2);
result = x;

for j=1:n
  [result(j), ifail] = s19ac(x(j));
end

disp('      x          ker(x)');
fprintf('%12.3e%12.3e\n',[x; result]);
```

9.2 Program Results

```
s19ac example results

      x          ker(x)
   1.000e-01   2.420e+00
   1.000e+00   2.867e-01
   2.500e+00  -6.969e-02
   5.000e+00  -1.151e-02
   1.000e+01   1.295e-04
   1.500e+01  -1.514e-08
```