NAG Toolbox

nag_specfun_gamma (s14aa)

1 Purpose

nag_specfun_gamma (s14aa) returns the value of the gamma function $\Gamma(x)$, via the function name.

2 Syntax

```
[result, ifail] = nag_specfun_gamma(x)
[result, ifail] = s14aa(x)
```

3 Description

nag_specfun_gamma (s14aa) evaluates an approximation to the gamma function $\Gamma(x)$. The function is based on the Chebyshev expansion:

$$\Gamma(1+u) = \sum_{r=0}^{\prime} a_r T_r(t), \qquad \text{where } 0 \leq u < 1, t = 2u - 1,$$

and uses the property $\Gamma(1+x) = x\Gamma(x)$. If $x = N + 1 + u$ where N is integral and $0 \leq u < 1$ then it follows that:

for $N > 0$, $\quad \Gamma(x) = (x-1)(x-2)\cdots(x-N)\Gamma(1+u)$,

for $N = 0$, $\quad \Gamma(x) = \Gamma(1+u)$,

for $N < 0$, $\quad \Gamma(x) = \dfrac{\Gamma(1+u)}{x(x+1)(x+2)\cdots(x-N-1)}$.

There are four possible failures for this function:

(i) if x is too large, there is a danger of overflow since $\Gamma(x)$ could become too large to be represented in the machine;

(ii) if x is too large and negative, there is a danger of underflow;

(iii) if x is equal to a negative integer, $\Gamma(x)$ would overflow since it has poles at such points;

(iv) if x is too near zero, there is again the danger of overflow on some machines. For small x, $\Gamma(x) \simeq 1/x$, and on some machines there exists a range of nonzero but small values of x for which $1/x$ is larger than the greatest representable value.

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** must not be zero or a negative integer.

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 $= 1$

The argument is too large. On softfailure the function returns the approximate value of $\Gamma(x)$ at the nearest valid argument.

ifail $= 2$

The argument is too large and negative. On softfailure the function returns zero.

ifail $= 3$ (*warning*)

The argument is too close to zero. On softfailure the function returns the approximate value of $\Gamma(x)$ at the nearest valid argument.

ifail $= 4$ (*warning*)

The argument is a negative integer, at which value $\Gamma(x)$ is infinite. On softfailure the function returns a large positive value.

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 δ and ϵ be the relative errors in the argument and the result respectively. If δ is somewhat larger than the ***machine precision*** (i.e., is due to data errors etc.), then ϵ and δ are approximately related by:

$$\epsilon \simeq |x\Psi(x)|\delta$$

(provided ϵ is also greater than the representation error). Here $\Psi(x)$ is the digamma function $\frac{\Gamma'(x)}{\Gamma(x)}$. Figure 1 shows the behaviour of the error amplification factor $|x\Psi(x)|$.

If δ is of the same order as ***machine precision***, then rounding errors could make ϵ slightly larger than the above relation predicts.

There is clearly a severe, but unavoidable, loss of accuracy for arguments close to the poles of $\Gamma(x)$ at negative integers. However relative accuracy is preserved near the pole at $x = 0$ right up to the point of failure arising from the danger of overflow.

Also accuracy will necessarily be lost as x becomes large since in this region

$$\epsilon \simeq \delta x \ln x.$$

However since $\Gamma(x)$ increases rapidly with x, the function must fail due to the danger of overflow before this loss of accuracy is too great. (For example, for $x = 20$, the amplification factor $\simeq 60$.)

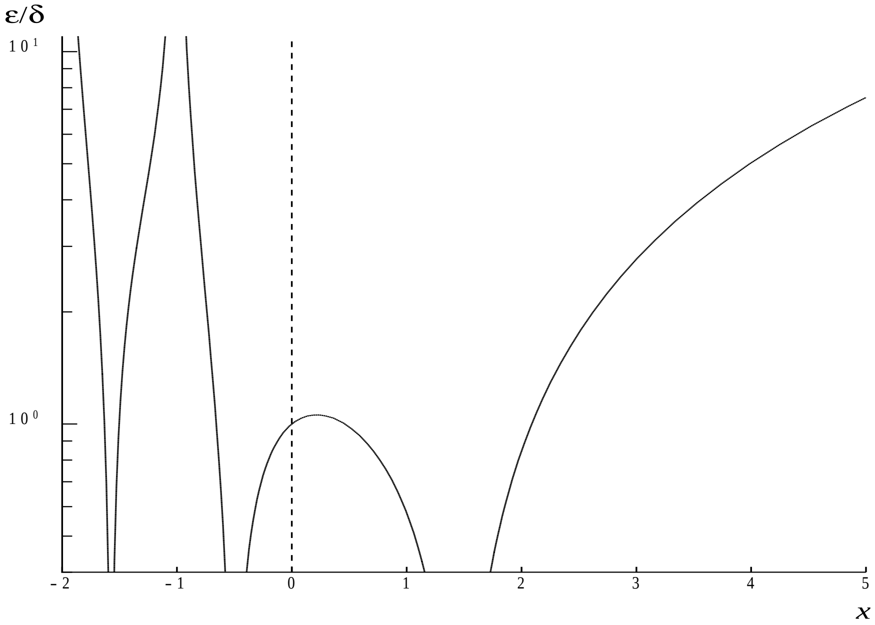


Figure 1

8 Further Comments

None.

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.

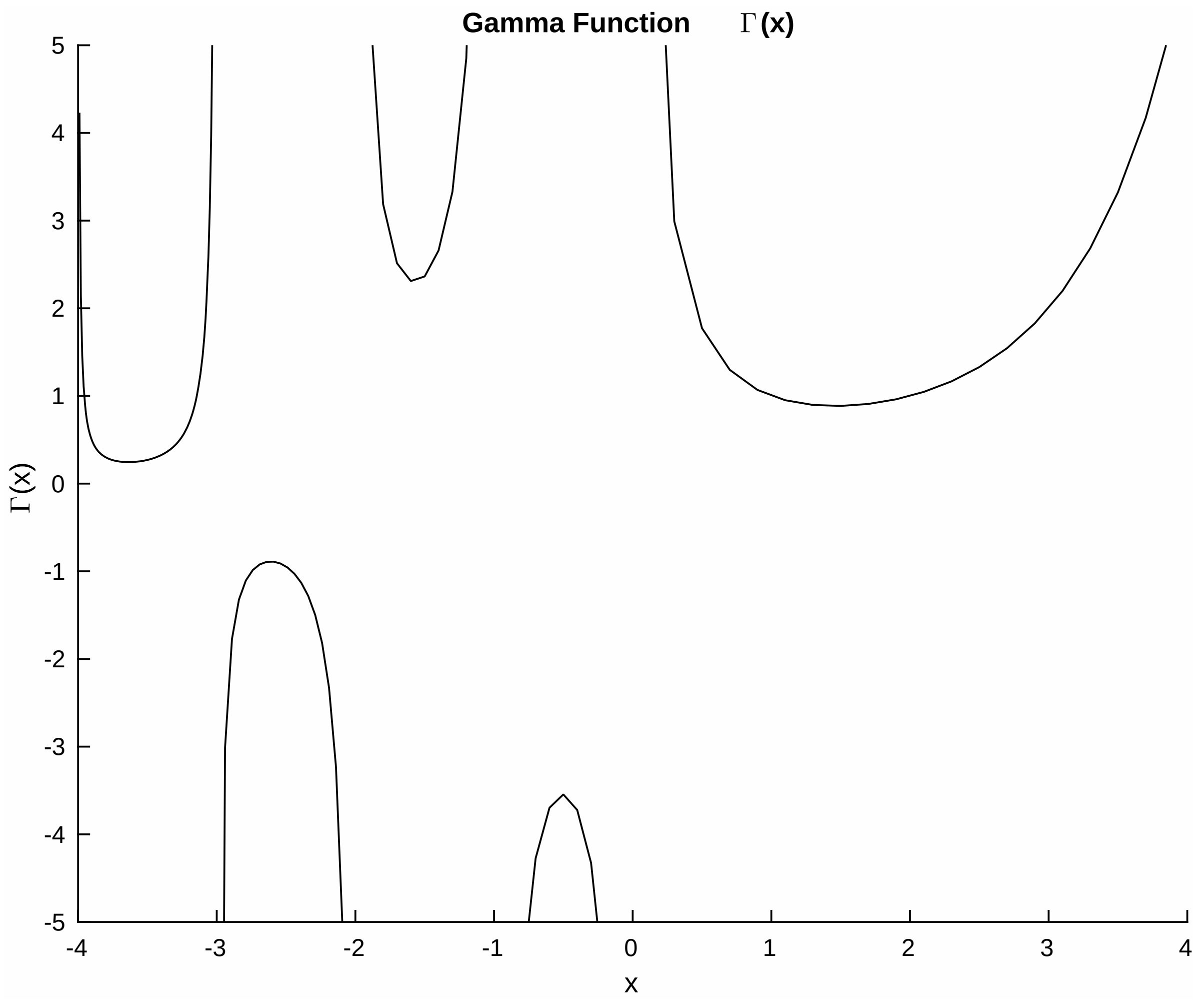
9.1 Program Text

```
     function s14aa_example

fprintf('s14aa example results\n\n');

x = [ 1    1.25     1.5     1.75     2       5       10     -1.5];
n = size(x,2);
result = x;

for j=1:n
  [result(j), ifail] = s14aa(x(j));
end

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

s14aa_plot;

function s14aa_plot
```

```
x = {[-3.99:0.01:-3.01]; [-2.99:0.05:-2.04]; [-1.9:0.1:-1.1];
 [-0.9:0.1:-0.1]; [0.1:0.2:3.9]};

for k = 1:5
  for j=1:numel(x{k})
    [g{k}(j), ifail] = s14aa(x{k}(j));
  end
end

fig1 = figure;
hold on;
for k = 1:5
  plot(x{k},g{k},'-r');
end
xlabel('x');
ylabel('\Gamma(x)');
title('Gamma Function \Gamma(x)');
axis([-4 4 -5 5]);
hold off;
% print(fig1,'-dpng','-r75','s14aa_fig1.png');
% print(fig1,'-deps','-r75','s14aa_fig1.eps');
```

9.2 Program Results

```
   s14aa example results

     x         Gamma(x)
 1.000e+00   1.000e+00
 1.250e+00   9.064e-01
 1.500e+00   8.862e-01
 1.750e+00   9.191e-01
 2.000e+00   1.000e+00
 5.000e+00   2.400e+01
 1.000e+01   3.629e+05
-1.500e+00   2.363e+00
```



Gamma Function Γ(x)
5
4
3
2
1
0
-1
-2
-3
-4
-5
Γ(x)
-4
-3
-2
-1
0
1
2
3
4
x