NAG Toolbox for MATLAB

s14aa

1 Purpose

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

2 Syntax

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

3 Description

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

$$\Gamma(1+u) = {\sum_{r=0}}' a_r T_r(t), \qquad \text{where } 0 \leq u < 1, t = 2u - 1,$$

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, $\Gamma(x) \simeq \frac{1}{x}$, 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 – double scalar**

The argument x of the function.

Constraint: **x** must not be zero or a negative integer.

5.2 Optional Input Parameters

None.

5.3 Input Parameters Omitted from the MATLAB Interface

None.

5.4 Output Parameters

1: **result – double scalar**

The result of the function.

2: **ifail – int32 scalar**

0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

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

ifail $= 2$

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

ifail $= 3$

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

ifail $= 4$

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

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 $\simeq 60$.)

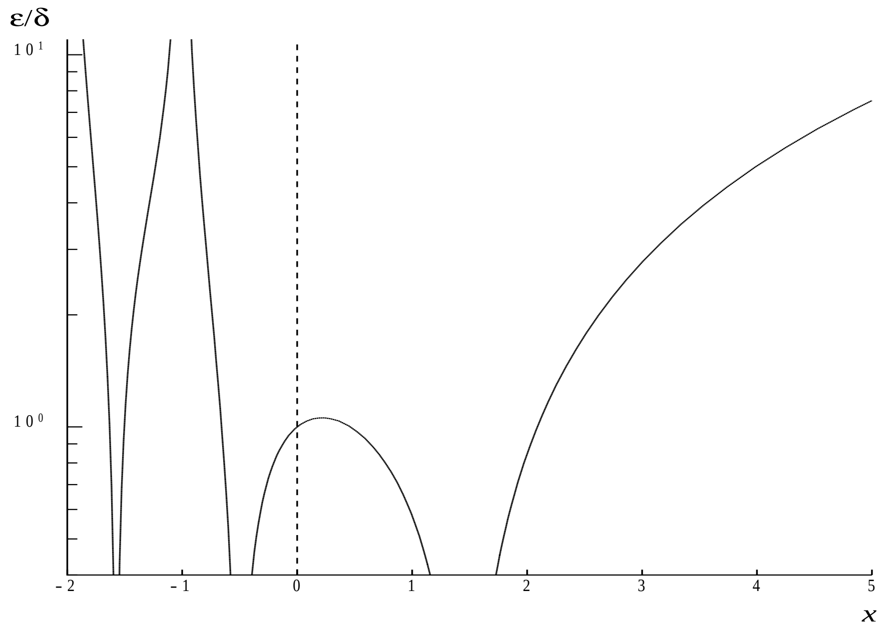

Figure 1

8 Further Comments

None.

9 Example

```
x = 1;
[result, ifail] = s14aa(x)
```

```
result =
     1
ifail =
           0
```