1 Purpose

s18ae returns the value of the modified Bessel Function $I_0(x)$, via the function name.

2 Syntax

\[ [\text{result}, \text{ifail}] = \text{s18ae}(x) \]

3 Description

s18ae evaluates an approximation to the modified Bessel Function of the first kind $I_0(x)$.

**Note:** $I_0(-x) = I_0(x)$, so the approximation need only consider $x \geq 0$.

The function is based on three Chebyshev expansions:

For $0 < x \leq 4$,

\[ I_0(x) = e^x \sum_{r=0}^{\infty} a_r T_r(t), \quad \text{where } t = 2 \left( \frac{x}{4} \right) - 1. \]

For $4 < x \leq 12$,

\[ I_0(x) = e^x \sum_{r=0}^{\infty} b_r T_r(t), \quad \text{where } t = \frac{x - 8}{4}. \]

For $x > 12$,

\[ I_0(x) = \frac{e^x}{\sqrt{x}} \sum_{r=0}^{\infty} c_r T_r(t), \quad \text{where } t = 2 \left( \frac{12}{x} \right) - 1. \]

For small $x$, $I_0(x) \approx 1$. This approximation is used when $x$ is sufficiently small for the result to be correct to machine precision.

For large $x$, the function must fail because of the danger of overflow in calculating $e^x$.

4 References


5 Parameters

5.1 Compulsory Input Parameters

1: \( x \) – double scalar
   
   The argument $x$ of the function.

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

6 Error Indicators and Warnings
Errors or warnings detected by the function:

ifail = 1
   x is too large. On soft failure the function returns the approximate value of $I_0(x)$ at the nearest
   valid argument.

7 Accuracy
Let $\delta$ and $\epsilon$ be the relative errors in the argument and result respectively.

If $\delta$ is somewhat larger than the machine precision (i.e., if $\delta$ is due to data errors etc.), then $\epsilon$ and $\delta$ are
approximately related by:

$$\epsilon \simeq \frac{|xI_1(x)|}{I_0(x)} \delta.$$  

Figure 1 shows the behaviour of the error amplification factor

$$\frac{|xI_1(x)|}{I_0(x)}.$$  

However if $\delta$ is of the same order as machine precision, then rounding errors could make $\epsilon$ slightly larger
than the above relation predicts.
For small $x$ the amplification factor is approximately $\frac{x^2}{2}$, which implies strong attenuation of the error, but in general $\epsilon$ can never be less than the machine precision.

For large $x$, $\epsilon \approx x\delta$ and we have strong amplification of errors. However the function must fail for quite moderate values of $x$, because $I_0(x)$ would overflow; hence in practice the loss of accuracy for large $x$ is not excessive. Note that for large $x$ the errors will be dominated by those of the standard function EXP.

### 8 Further Comments

None.

### 9 Example

```matlab
x = 0;
[result, ifail] = s18ae(x)
```

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