nag_airy_ai (s17agc)

1. Purpose

nag_airy_ai (s17agc) returns a value for the Airy function $\mathrm{Ai}(x)$.

2. Specification

```
#include <nag.h>
#include <nags.h>

double nag_airy_ai(double x, NagError *fail)
```

3. Description

This function evaluates an approximation to the Airy function, $\mathrm{Ai}(x)$. It is based on a number of Chebyshev expansions.

For large negative arguments, it is impossible to calculate the phase of the oscillatory function with any precision and so the function must fail. This occurs if $x < -(3/2\epsilon)^{2/3}$, where ϵ is the ***machine precision***.

For large positive arguments, where Ai decays in an essentially exponential manner, there is a danger of underflow so the function must fail.

4. Parameters

x
Input: the argument x of the function.

fail
The NAG error parameter, see the Essential Introduction to the NAG C Library.

5. Error Indications and Warnings

NE_REAL_ARG_GT
On entry, **x** must not be greater than ⟨*value*⟩: **x** = ⟨*value*⟩.
x is too large and positive. The function returns zero.

NE_REAL_ARG_LT
On entry, **x** must not be less than ⟨*value*⟩: **x** = ⟨*value*⟩.
x is too large and negative. The function returns zero.

6. Further Comments

6.1. Accuracy

For negative arguments the function is oscillatory and hence absolute error is the appropriate measure. In the positive region the function is essentially exponential-like and here relative error is appropriate. The absolute error, E, and the relative error, ϵ, are related in principle to the relative error in the argument, δ, by $E \simeq |x\mathrm{Ai}'(x)|\,\delta$, $\epsilon \simeq |x\mathrm{Ai}'(x)/\mathrm{Ai}(x)|\,\delta$.

In practice, approximate equality is the best that can be expected. When δ, ϵ or E is of the order of the ***machine precision***, the errors in the result will be somewhat larger.

For small x, errors are strongly damped by the function and hence will be bounded by the ***machine precision***.

For moderate negative x, the error behaviour is oscillatory but the amplitude of the error grows like amplitude $(E/\delta) \sim |x|^{5/4}/\sqrt{\pi}$. However the phase error will be growing roughly like $2\sqrt{|x|^3}/3$ and hence all accuracy will be lost for large negative arguments due to the difficulty in calculating sin and cos to any accuracy if $2\sqrt{|x|^3}/3 > 1/\delta$.

For large positive arguments, the relative error amplification is considerable, $\epsilon/\delta \sim \sqrt{x^3}$.

This means a loss of roughly two decimal places accuracy for arguments in the region of 20. However, very large arguments are not possible due to the danger of setting underflow, and so the errors are limited in practice.

6.2. References

Abramowitz M and Stegun I A (1968) *Handbook of Mathematical Functions* Dover Publications, New York ch 10.4 p 446.

7. See Also

nag_airy_ai_deriv (s17ajc)

8. Example

The following program reads values of the argument x from a file, evaluates the function at each value of x and prints the results.

8.1. Program Text

```
/* nag_airy_ai(s17agc) Example Program
 *
 * Copyright 1990 Numerical Algorithms Group.
 *
 * Mark 2 revised, 1992.
 */

#include <nag.h>
#include <stdio.h>
#include <nag_stdlib.h>
#include <nags.h>

main()
{
  double x, y;

  /* Skip heading in data file */
  Vscanf("%*[^\n]");
  Vprintf("s17agc Example Program Results\n");
  Vprintf("     x           y\n");
  while (scanf("%lf", &x) != EOF)
    {
      y = s17agc(x, NAGERR_DEFAULT);
      Vprintf("%12.3e%12.3e\n", x, y);
    }
  exit(EXIT_SUCCESS);
}
```

8.2. Program Data

```
s17agc Example Program Data
             -10.0
             -1.0
             0.0
             1.0
             5.0
             10.0
             20.0
```

8.3. Program Results

```
s17agc Example Program Results
     x           y
  -1.000e+01   4.024e-02
  -1.000e+00   5.356e-01
   0.000e+00   3.550e-01
   1.000e+00   1.353e-01
   5.000e+00   1.083e-04
   1.000e+01   1.105e-10
   2.000e+01   1.692e-27
```