nag_erfc (s15adc)

1. Purpose

nag_erfc (s15adc) returns the value of the complementary error function, $\mathrm{erfc}\, x$.

2. Specification

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

double nag_erfc(double x)
```

3. Description

The function calculates an approximate value for the complement of the error function

$$\mathrm{erfc}\, x = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-u^2} du = 1 - \mathrm{erf} x.$$

The approximation is based on a Chebyshev expansion.

4. Parameters

x
Input: the argument x of the function.

5. Error Indications and Warnings

None.

6. Further Comments

6.1. Accuracy

If δ and ϵ are relative errors in the argument and result, respectively, then in principle $|\epsilon| \simeq |(2xe^{-x^2}/\sqrt{\pi}\mathrm{erfc}\, x)\, \delta|$, so that the relative error in the argument, x, is amplified by a factor $(2xe^{-x^2})/(\sqrt{\pi}\mathrm{erfc}\, x)$ in the result.

Near $x = 0$ this factor behaves as $2x/\sqrt{\pi}$ and hence the accuracy is largely determined by the ***machine precision***. Also for large negative x, where the factor is $\sim xe^{-x^2}/\sqrt{\pi}$, accuracy is mainly limited by ***machine precision***. However, for large positive x, the factor becomes $\sim 2x^2$ and to an extent relative accuracy is necessarily lost. The absolute accuracy E is given by $E \simeq (2xe^{-x^2}/\sqrt{\pi})\delta$ so absolute accuracy is guaranteed for all x.

6.2. References

Abramowitz M and Stegun I A (1968) *Handbook of Mathematical Functions* Dover Publications, New York ch 7 p 297.

7. See Also

nag_erf (s15aec)

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_erfc(s15adc) Example Program
 *
 * Copyright 1990 Numerical Algorithms Group.
 *
 * Mark 1, 1990.
 */

#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("s15adc Example Program Results\n");
  Vprintf("     x           y\n");
  while (scanf("%lf", &x) != EOF)
    {
      y = s15adc(x);
      Vprintf("%12.3e%12.3e\n", x, y);
    }
  exit(EXIT_SUCCESS);
}
```

8.2. Program Data

```
s15adc Example Program Data
           -10.0
            -1.0
             0.0
             1.0
            10.0
```

8.3. Program Results

```
s15adc Example Program Results
     x           y
  -1.000e+01   2.000e+00
  -1.000e+00   1.843e+00
   0.000e+00   1.000e+00
   1.000e+00   1.573e-01
   1.000e+01   2.088e-45
```