nag_cumul_normal_complem (s15acc)

1. Purpose

nag_cumul_normal_complem (s15acc) returns the value of the complement of the cumulative normal distribution function $Q(x)$.

2. Specification

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

double nag_cumul_normal_complem(double x)
```

3. Description

The function evaluates an approximate value for the complement of the cumulative normal distribution function

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-u^2/2}\, du.$$

The function is based on the fact that

$$Q(x) = \tfrac{1}{2}\,\mathrm{erfc}(x/\sqrt{2}).$$

4. Parameters

x

Input: the argument x of the function.

5. Error Indications and Warnings

None.

6. Further Comments

6.1. Accuracy

If ϵ and δ are the relative errors in result and argument, respectively, then in principle they are related by $|\epsilon| \simeq |(xe^{-x^2/2}/\sqrt{2\pi}Q(x))\,\delta|$.

For x negative or small positive the multiplying factor is always less than one and accuracy is mainly limited by ***machine precision***. For large positive x we find $\epsilon \sim x^2\delta$ and hence to a certain extent relative accuracy is unavoidably lost. However the absolute error in the result, E, is given by $|E| \simeq |(xe^{-x^2/2}/\sqrt{2\pi})\,\delta|$, and since this multiplying factor is always less than one absolute accuracy can be guaranteed for all x.

6.2. References

Abramowitz M and Stegun I A (1968) *Handbook of Mathematical Functions* Dover Publications, New York ch 7.1 p 297 and ch 26.2 p 931.

7. See Also

nag_deviates_normal_dist (g01cec)
nag_cumul_normal (s15abc)

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_cumul_normal_complem(s15acc) Example Program
 *
 * Copyright 1990 Numerical Algorithms Group.
 *
 * Mark 1, 1990.
 *
 * Mark 3 revised, 1994.
 */

#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("s15acc Example Program Results\n");
  Vprintf("     x           y\n");
  while (scanf("%lf", &x) != EOF)
    {
      y = s15acc(x);
      Vprintf("%12.3e%12.3e\n", x, y);
    }
  exit(EXIT_SUCCESS);
}
```

8.2. Program Data

```
s15acc Example Program Data
           -20.0
            -1.0
             0.0
             1.0
             2.0
            20.0
```

8.3. Program Results

```
s15acc Example Program Results
     x           y
  -2.000e+01   1.000e+00
  -1.000e+00   8.413e-01
   0.000e+00   5.000e-01
   1.000e+00   1.587e-01
   2.000e+00   2.275e-02
   2.000e+01   2.754e-89
```