nag_fresnel_s (s20acc) returns a value for the Fresnel integral \( S(x) \).

The function is based on three Chebyshev expansions:

For \( 0 < x \leq 3 \),
\[
    S(x) = x^3 \sum_{r=0} a_r T_r(t), \quad \text{with} \ t = 2 \left( \frac{x}{3} \right)^4 - 1.
\]

For \( x > 3 \),
\[
    S(x) = \frac{1}{2} - \frac{f(x)}{x} \cos \left( \frac{x^{\pi/2}}{2} \right) - \frac{g(x)}{x^3} \sin \left( \frac{x^{\pi/2}}{2} \right),
\]
where \( f(x) = \sum_{r=0} b_r T_r(t) \), and \( g(x) = \sum_{r=0} c_r T_r(t) \), with \( t = 2 \left( \frac{3}{x} \right)^4 - 1 \).

For small \( x \), \( S(x) \approx \frac{\pi}{6} x^3 \). This approximation is used when \( x \) is sufficiently small for the result to be correct to \emph{machine precision}. For very small \( x \), this approximation would underflow; the result is then set exactly to zero.

For large \( x \), \( f(x) \approx \frac{1}{x} \) and \( g(x) \approx \frac{1}{x^3} \). Therefore for moderately large \( x \), when \( \frac{1}{x^3} \) is negligible compared with \( \frac{1}{x} \), the second term in the approximation for \( x > 3 \) may be dropped. For very large \( x \), when \( \frac{1}{x^3} \) becomes negligible, \( S(x) \approx \frac{1}{2} \). However there will be considerable difficulties in calculating \( \cos \left( \frac{x^{\pi/2}}{2} \right) \) accurately before this final limiting value can be used. Since \( \cos \left( \frac{x^{\pi/2}}{2} \right) \) is periodic, its value is essentially determined by the fractional part of \( x^2 \). If \( x^2 = N + \theta \) where \( N \) is an integer and \( 0 \leq \theta < 1 \), then \( \cos \left( \frac{x^{\pi/2}}{2} \right) \) depends on \( \theta \) and on \( N \) modulo 4. By exploiting this fact, it is possible to retain
significance in the calculation of $\cos\left(\frac{\pi}{2}x^2\right)$ either all the way to the very large $x$ limit, or at least until the integer part of $\frac{x}{2}$ is equal to the maximum integer allowed on the machine.

4 References

5 Arguments
1: $x$ – double
   *Input*
   
   On entry: the argument $x$ of the function.

6 Error Indicators and Warnings
None.

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:

$$
\epsilon \approx \left| \frac{x \sin\left(\frac{\pi}{2}x^2\right)}{S(x)} \right| \delta.
$$

Figure 1 shows the behaviour of the error amplification factor $\left| \frac{x \sin\left(\frac{\pi}{2}x^2\right)}{S(x)} \right|$.

However if $\delta$ is of the same order as the *machine precision*, then rounding errors could make $\epsilon$ slightly larger than the above relation predicts.

For small $x$, $\epsilon \approx 3\delta$ and hence there is only moderate amplification of relative error. Of course for very small $x$ where the correct result would underflow and exact zero is returned, relative error-control is lost.

For moderately large values of $x$,

$$
|\epsilon| \approx \left| 2x \sin\left(\frac{\pi}{2}x^2\right) \right| |\delta|,
$$

and the result will be subject to increasingly large amplification of errors. However the above relation breaks down for large values of $x$ (i.e., when $\frac{1}{x^2}$ is of the order of the *machine precision*); in this region the relative error in the result is essentially bounded by $\frac{2}{\pi x}$.

Hence the effects of error amplification are limited and at worst the relative error loss should not exceed half the possible number of significant figures.
8 Parallelism and Performance
Not applicable.

9 Further Comments
None.

10 Example
This example reads values of the argument \( x \) from a file, evaluates the function at each value of \( x \) and prints the results.

10.1 Program Text
/* nag_fresnel_s (s20acc) Example Program.  *
* Copyright 2014 Numerical Algorithms Group.  *
* Mark 2 revised, 1992.  */

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

int main(void)
{
    Integer exit_status = 0;
    double x, y;

    /* Skip heading in data file */
    #ifdef _WIN32
    scanf_s("%*[\n\n]");
    #else
    scanf("%*[\n\n]");
    #endif

printf("nag_fresnel_s (s20acc) Example Program Results\n");
printf(" x y\n");
#endif
while (scanf_s("%lf", &x) != EOF)
#else
while (scanf("%lf", &x) != EOF)
#endif
{
    /* nag_fresnel_s (s20acc).
     * Fresnel integral S(x)
     */
    y = nag_fresnel_s(x);
    printf("%12.3e%12.3e\n", x, y);
}
return exit_status;

10.2 Program Data
nag_fresnel_s (s20acc) Example Program Data
0.0
0.5
1.0
2.0
4.0
5.0
6.0
8.0
10.0
-1.0
1000.0

10.3 Program Results
nag_fresnel_s (s20acc) Example Program Results
 x      y
0.000e+00 0.000e+00
5.000e-01 6.473e-02
1.000e+00 4.383e-01
2.000e+00 3.434e-01
4.000e+00 4.205e-01
5.000e+00 4.992e-01
6.000e+00 4.470e-01
8.000e+00 4.602e-01
1.000e+01 4.682e-01
-1.000e+00 -4.383e-01
1.000e+03 4.997e-01
Example Program
Returns a Value for the Fresnel Integral \( S(x) \)