G05CAF – NAG Fortran Library Routine Document

Note. Before using this routine, please read the Users' Note for your implementation to check the interpretation of bold italicised terms and other implementation-dependent details.

1 Purpose

G05CAF returns a pseudo-random number taken from a uniform distribution between 0 and 1.

2 Specification

```
real FUNCTION G05CAF(X)
real                X
```

3 Description

This routine returns the next pseudo-random number from the basic uniform (0,1) generator.

The basic generator uses a multiplicative congruential algorithm:

$$b_{i+1} = 13^{13} \times b_i \bmod 2^{59}$$

The integer b_{i+1} is divided by 2^{59} to yield a ***real*** value y, which is guaranteed to satisfy

$$0 < y < 1$$

The value of b_i is saved internally in the code. The initial value b_0 is set by default to $123456789 \times (2^{32}+1)$, but the sequence may be re-initialized by a call to G05CBF (for a repeatable sequence) or G05CCF (for a non-repeatable sequence). The current value of b_i may be saved by a call to G05CFF, and restored by a call to G05CGF.

G05FAF may be used to generate a vector of n pseudo-random numbers which are exactly the same as n successive values of G05CAF. On vector-processing machines G05FAF is likely to be much faster.

4 References

[1] Knuth D E (1981) *The Art of Computer Programming (Volume 2)* Addison–Wesley (2nd Edition)

5 Parameters

1: X — ***real*** *Dummy*

A dummy argument (originally required by ANSI Fortran 66 syntax).

6 Error Indicators and Warnings

None.

7 Accuracy

Not applicable.

8 Further Comments

The period of the basic generator is 2^{57}.

Its performance has been analysed by the Spectral Test, see Knuth [1], Section 3.3.4, yielding the following results in the notation of Knuth [1].

n	ν_n	Upper bound for ν_n
2	3.44×10^8	4.08×10^8
3	4.29×10^5	5.88×10^5
4	1.72×10^4	2.32×10^4
5	1.92×10^3	3.33×10^3
6	593	939
7	198	380
8	108	197
9	67	120

The right-hand column gives an upper bound for the values of ν_n attainable by any multiplicative congruential generator working modulo 2^{59}.

An informal interpretation of the quantities ν_n is that consecutive n-tuples are statistically uncorrelated to an accuracy of $1/\nu_n$. This is a theoretical result; in practice the degree of randomness is usually much greater than the above figures might support. More details are given in Knuth [1], and in the references cited therein.

Note that the achievable accuracy drops rapidly as the number of dimensions increases. This is a property of all multiplicative congruential generators and is the reason why very long periods are needed even for samples of only a few random numbers.

9 Example

The example program prints the first five pseudo-random numbers from a uniform distribution between 0 and 1, generated by G05CAF after initialisation by G05CBF.

9.1 Program Text

Note. The listing of the example program presented below uses bold italicised terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     G05CAF Example Program Text
*     Mark 14 Revised.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          NOUT
      PARAMETER        (NOUT=6)
*     .. Local Scalars ..
      real             X
      INTEGER          I
*     .. External Functions ..
      real             G05CAF
      EXTERNAL         G05CAF
*     .. External Subroutines ..
      EXTERNAL         G05CBF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G05CAF Example Program Results'
      WRITE (NOUT,*)
      CALL G05CBF(0)
      DO 20 I = 1, 5
*
         X = G05CAF(X)
*
         WRITE (NOUT,99999) X
   20 CONTINUE
      STOP
*
99999 FORMAT (1X,F10.4)
      END
```

9.2 Program Data

None.

9.3 Program Results

```
G05CAF Example Program Results

    0.7951
    0.2257
    0.3713
    0.2250
    0.8787
```