

NAG Library Routine Document

G05LGF

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

G05LGF generates a vector of pseudorandom numbers uniformly distributed over the interval $[a, b]$.

2 Specification

```
SUBROUTINE G05LGF(A, B, N, X, IGEN, ISEED, IFAIL)
INTEGER          N, IGEN, ISEED(4), IFAIL
double precision A, B, X(N)
```

3 Description

If $a = 0$ and $b = 1$, G05LGF returns the next n values y_i from a uniform $(0, 1)$ generator (see G05KAF for details).

For other values of a and b , G05LGF applies the transformation

$$x_i = a + (b - a)y_i.$$

The routine ensures that the values x_i lie in the closed interval $[a, b]$.

If computing sequentially and using the same generator, G05LGF always generates exactly the same pseudorandom numbers as would n consecutive calls of G05KAF and on many machines is likely to be much faster.

One of the initialization routines G05KBF (for a repeatable sequence if computed sequentially) or G05KCF (for a non-repeatable sequence) must be called prior to the first call to G05LGF.

4 References

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

5 Parameters

- | | | |
|----|---|---------------|
| 1: | A – <i>double precision</i> | <i>Input</i> |
| 2: | B – <i>double precision</i> | <i>Input</i> |
| | <i>On entry:</i> the end points a and b of the uniform distribution. | |
| | <i>Constraint:</i> $A \leq B$. | |
| 3: | N – INTEGER | <i>Input</i> |
| | <i>On entry:</i> n , the number of pseudorandom numbers to be generated. | |
| | <i>Constraint:</i> $N \geq 0$. | |
| 4: | $X(N)$ – <i>double precision</i> array | <i>Output</i> |
| | <i>On exit:</i> the n pseudorandom numbers from the specified uniform distribution. | |

- 5: IGEN – INTEGER *Input*
On entry: must contain the identification number for the generator to be used to return a pseudorandom number and should remain unchanged following initialization by a prior call to G05KBF or G05KCF.
- 6: ISEED(4) – INTEGER array *Communication Array*
On entry: contains values which define the current state of the selected generator.
On exit: contains updated values defining the new state of the selected generator.
- 7: IFAIL – INTEGER *Input/Output*
On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction for details.
On exit: IFAIL = 0 unless the routine detects an error (see Section 6).
 For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, if you are not familiar with this parameter, the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

6 Error Indicators and Warnings

If on entry IFAIL = 0 or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL = 1

On entry, $N < 0$.

IFAIL = 2

On entry, $B < A$.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

This example prints five pseudorandom numbers from a uniform distribution between -1.0 and 1.0, generated by a single call to G05LGF, after initialization by G05KBF.

9.1 Program Text

```
*      G05LGF Example Program Text
*      Mark 20 Release. NAG Copyright 2001.
*      .. Parameters ..
      INTEGER          NOUT, N
      PARAMETER        (NOUT=6,N=5)
*      .. Local Scalars ..
      DOUBLE PRECISION A, B
      INTEGER          IFAIL, IGEN
*      .. Local Arrays ..
```

```

      DOUBLE PRECISION X(N)
      INTEGER          ISEED(4)
*      .. External Subroutines ..
      EXTERNAL          G05KBF, G05LGF
*      .. Executable Statements ..
      WRITE (NOUT,*) 'G05LGF Example Program Results'
      WRITE (NOUT,*)
*      Initialize the seed to a repeatable sequence
      ISEED(1) = 1762543
      ISEED(2) = 9324783
      ISEED(3) = 42344
      ISEED(4) = 742355
*      IGEN identifies the stream.
      IGEN = 1
      CALL G05KBF(IGEN,ISEED)
*
      A = -1.0D0
      B = 1.0D0
      IFAIL = 1
      CALL G05LGF(A,B,N,X,IGEN,ISEED,IFAIL)
*
      IF (IFAIL.EQ.0) THEN
*
          WRITE (NOUT,99999) X
*
      ELSE
          WRITE (NOUT,99998) IFAIL
      END IF
*
99999 FORMAT (1X,F10.4)
99998 FORMAT (1X,' ** G05LGF returned with IFAIL = ',I5)
      END

```

9.2 Program Data

None.

9.3 Program Results

G05LGF Example Program Results

```

-0.8214
 0.9019
-0.1872
 0.4864
 0.8995

```
