NAG Library Routine Document

G05KGF

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

G05KGF initializes the selected base generator to generate a non-repeatable sequence of variates. The base generator can then be used by the group of pseudorandom number routines (see G05KHF–G05KJF, G05NCF, G05NDF, G05PDF–G05PJF, G05PXF–G05PZF, G05RCF, G05RDF, G05RYF, G05RZF and G05SAF–G05TLF) and the quasi-random scrambled sequence initialization routine, G05YNF.

2 Specification

```
SUBROUTINE G05KGF(GENID, SUBID, STATE, LSTATE, IFAIL)
INTEGER           GENID, SUBID, STATE(LSTATE), LSTATE, IFAIL
```

3 Description

G05KGF selects a base generator through the input value of the parameters GENID and SUBID, and then initializes it based on the values taken from the real-time clock, resulting in the same base generator yielding different sequences of random numbers each time the calling program is run. It should be noted that there is no guarantee of statistical properties between sequences, only within sequences.

A definition of some of the terms used in this description, along with details of the various base generators can be found in the G05 Chapter Introduction.

4 References

None.

5 Parameters

1: GENID – INTEGER *Input*

On entry: must contain the type of base generator to use.

GENID = 1
NAG basic generator.

GENID = 2
Wichmann Hill I generator.

GENID = 3
Mersenne Twister.

GENID = 4
Wichmann Hill II generator.

GENID = 5
ACORN generator.

See the G05 Chapter Introduction for details of each of the base generators.

Constraint: GENID = 1, 2, 3, 4 or 5.

2: SUBID – INTEGER *Input*

On entry: if GENID $= 2$, SUBID indicates which of the 273 sub-generators to use. In this case, the $((|\text{SUBID}| + 272)\ mod\ 273) + 1$ sub-generator is used.

If GENID $= 5$, SUBID indicates the values of k and p to use, where k is the order of the generator, and p controls the size of the modulus, M, with $M = 2^{(p \times 30)}$. If SUBID < 1, the default values of $k = 10$ and $p = 2$ are used, otherwise values for k and p are calculated from the formula, SUBID $= k + 1000p$.

For all other values of GENID, SUBID is not referenced.

3: STATE(LSTATE) – INTEGER array *Communication Array*

On exit: contains information on the selected base generator and its current state.

4: LSTATE – INTEGER *Input/Output*

On entry: the dimension of the STATE array, or a value < 1.

On exit: if LSTATE < 1 on entry, then the required length of the STATE array for the chosen base generator, otherwise LSTATE is unchanged.

Constraints:

if GENID $= 1$, LSTATE ≥ 17;
if GENID $= 2$, LSTATE ≥ 21;
if GENID $= 3$, LSTATE ≥ 633;
if GENID $= 4$, LSTATE ≥ 29;
if GENID $= 5$, LSTATE $\geq (k + 1) \times p + 12$, where k and p are defined by SUBID;
otherwise LSTATE < 1.

5: 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 $= -1$

Minimum length for STATE array returned in parameter LSTATE but STATE array not initialized.

IFAIL $= 1$

On entry, GENID $\neq 1$, 2, 3, 4 or 5.

IFAIL $= 2$

Invalid SUBID.

IFAIL = 4

On entry, LSTATE > 0 and LSTATE is too small for specified base generator.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

This example prints the first five pseudorandom real numbers from a uniform distribution between 0 and 1, generated by G05SAF after initialization by G05KGF.

9.1 Program Text

```
*     G05KGF Example Program Text
*     Mark 22 Release. NAG Copyright 2006.
*     .. Parameters ..
      INTEGER          NOUT
      PARAMETER        (NOUT=6)
      INTEGER          MSTATE, N
      PARAMETER        (MSTATE=633,N=5)
*     .. Local Scalars ..
      INTEGER          GENID, I, IFAIL, LSTATE, SUBID
*     .. Local Arrays ..
      DOUBLE PRECISION X(N)
      INTEGER          STATE(MSTATE)
*     .. External Subroutines ..
      EXTERNAL         G05KGF, G05SAF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G05KGF Example Program Results'
      WRITE (NOUT,*)
*     GENID and SUBID identify the base generator
      GENID = 1
      SUBID = 1
*     Initialize the generator to an unrepeatable sequence
      LSTATE = MSTATE
      IFAIL = 1
      CALL G05KGF(GENID,SUBID,STATE,LSTATE,IFAIL)
      IF (IFAIL.NE.0) THEN
         WRITE (NOUT,99998) IFAIL
         GO TO 20
      END IF
*     Generate the variates
      IFAIL = 1
      CALL G05SAF(N,STATE,X,IFAIL)
      IF (IFAIL.NE.0) THEN
         WRITE (NOUT,99997) IFAIL
         GO TO 20
      END IF
*     Display the variates
      WRITE (NOUT,99999) (X(I),I=1,N)
*
   20 CONTINUE
*
99999 FORMAT (1X,F10.4)
99998 FORMAT (1X,' ** G05KGF returned with IFAIL = ',I5)
99997 FORMAT (1X,' ** G05SAF returned with IFAIL = ',I5)
      END
```

9.2 Program Data

None.

9.3 Program Results

```
 G05KGF Example Program Results

     0.0629
     0.6424
     0.9855
     0.7981
     0.5724
```