NAG Fortran Library Routine Document

G05NAF

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

G05NAF performs a pseudo-random permutation of a vector of integers.

2 Specification

```
SUBROUTINE G05NAF(INDEX, N, IGEN, ISEED, IFAIL)
INTEGER          INDEX(N), N, IGEN, ISEED(4), IFAIL
```

3 Description

G05NAF permutes the elements of an integer array without inspecting their values. Each of the $n!$ possible permutations of the n values may be regarded as being equally probable.

Even for modest values of n (greater than 25 say), it is theoretically impossible that all $n!$ permutations may occur, as $n!$ exceeds the cycle length of G05KAF for any valid value of IGEN. For practical purposes this is irrelevant, as the time necessary to generate all possible permutations is many millenia.

One of the initialisation 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 G05NAF.

4 References

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

Kendall M G and Stuart A (1969) *The Advanced Theory of Statistics (Volume 1)* (3rd Edition) Griffin

5 Parameters

1:    INDEX(N) – INTEGER array   *Input/Output*

On entry: the n integer values to be permuted.

On exit: the n permuted integer values.

2:    N – INTEGER   *Input*

On entry: the number of values to be permuted.

Constraint: N $\geq$ 1.

3:    IGEN – INTEGER   *Input*

On entry: must contain the identification number for the generator to be used to return a pseudo-random number and should remain unchanged following initialisation by a prior call to one of the routines G05KBF or G05KCF.

4:    ISEED(4) – INTEGER array   *Input/Output*

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.

5:     IFAIL – INTEGER   *Input/Output*

On entry: IFAIL must be set to 0, −1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 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, for users 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 < 1.

7   Accuracy

Not applicable.

8   Further Comments

None.

9   Example

In the example program a vector containing the first eight positive integers in ascending order is permuted by a call to G05NAF and the permutation is printed. This is repeated a total of ten times, after initialisation by G05KBF.

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.

```
*      G05NAF Example Program Text
*      Mark 20 Release. NAG Copyright 2001.
*      .. Parameters ..
       INTEGER          NOUT, N
       PARAMETER        (NOUT=6,N=8)
*      .. Local Scalars ..
       INTEGER          I, IFAIL, IGEN, J, K, M
*      .. Local Arrays ..
       INTEGER          INDEX(N), ISEED(4)
*      .. External Subroutines ..
       EXTERNAL         G05KBF, G05NAF
*      .. Executable Statements ..
       WRITE (NOUT,*) 'G05NAF Example Program Results'
       WRITE (NOUT,*)
       M = 10
*      Initialise 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)
      WRITE (NOUT,99998) M, ' Permutations of first ', N, ' integers'
      WRITE (NOUT,*)
      DO 40 J = 1, M
         DO 20 I = 1, N
            INDEX(I) = I
  20     CONTINUE
         IFAIL = 0
         CALL G05NAF(INDEX,N,IGEN,ISEED,IFAIL)
*
         WRITE (NOUT,99999) (INDEX(K),K=1,N)
  40 CONTINUE
      STOP
*
99999 FORMAT (1X,8I3)
99998 FORMAT (1X,I2,A,I1,A)
      END
```

9.2 Program Data

None.

9.3 Program Results

```
 G05NAF Example Program Results

 10 Permutations of first 8 integers

    2   4   3   7   8   6   5   1
    2   3   6   4   7   5   1   8
    6   5   1   4   7   8   3   2
    7   3   2   5   1   4   8   6
    7   4   8   5   6   2   3   1
    2   1   5   8   4   3   7   6
    7   3   4   6   2   5   1   8
    4   1   7   5   6   8   3   2
    3   1   5   2   8   7   4   6
    8   7   6   1   3   2   5   4
```