NAG Fortran Library Routine Document

G05NBF

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

G05NBF selects a pseudo-random sample without replacement from an integer vector.

2 Specification

```
SUBROUTINE G05NBF(IPOP, N, ISAMPL, M, IGEN, ISEED, IFAIL)
INTEGER           IPOP(N), N, ISAMPL(M), M, IGEN, ISEED(4), IFAIL
```

3 Description

G05NBF The routine selects m elements from a population vector IPOP of length n and places them in a sample vector ISAMPL. Their order in IPOP will be preserved in ISAMPL. Each of the $\binom{n}{m}$ possible combinations of elements of ISAMPL may be regarded as being equally probable.

For moderate or large values of n (greater than 75 say), it is theoretically impossible that all combinations of size m may occur, unless m is near 1 or near n. This is because $\binom{n}{m}$ exceeds the cycle length of G05KAF for all valid values of IGEN. For practical purposes this is irrelevant, as the time taken to generate all possible combinations 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 G05NBF.

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: IPOP(N) – INTEGER array *Input*

On entry: the population to be sampled.

2: N – INTEGER *Input*

On entry: the number of elements in the population vector to be sampled.

Constraint: N $\geq$ 1.

3: ISAMPL(M) – INTEGER array *Output*

On entry: the selected sample.

4: M – INTEGER *Input*

On entry: the sample size.

Constraint: 1 $\leq$ M $\leq$ N.

5: 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.

6: 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.

7: 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.

IFAIL $= 2$

On entry, M < 1,
or M $>$ N.

7 Accuracy

Not applicable.

8 Further Comments

The time taken by the routine is of order n.

In order to sample other kinds of vectors, or matrices of higher dimension, the following technique may be used:

(a) set IPOP$(i) = i$, for $i = 1, 2, \ldots, n$;

(b) use G05NBF to take a sample from IPOP and put it into ISAMPL;

(c) use the contents of ISAMPL as a set of indices to access the relevant vector or matrix.

In order to divide a population into several groups, G05NAF is more efficient.

9 Example

In the example program random samples of size $1, 2, \ldots, 8$ are selected from a vector containing the first eight positive integers in ascending order. The samples are generated and printed for each sample size by a call to G05NBF 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.

```
*     G05NBF Example Program Text
*     Mark 20 Release. NAG Copyright 2001.
*     .. Parameters ..
      INTEGER          NOUT, N
      PARAMETER        (NOUT=6,N=8)
*     .. Local Scalars ..
      INTEGER          I, IFAIL, IGEN, K, M
*     .. Local Arrays ..
      INTEGER          IPOP(N), ISAMPL(N), ISEED(4)
*     .. External Subroutines ..
      EXTERNAL         G05KBF, G05NBF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G05NBF Example Program Results'
      WRITE (NOUT,*)
*     Initialise the seed to a repeatable sequence
      ISEED(1) = 1762543
      ISEED(2) = 9324783
      ISEED(3) = 1542344
      ISEED(4) = 742355
*     IGEN identifies the stream.
      IGEN = 1
      CALL G05KBF(IGEN,ISEED)
      WRITE (NOUT,99999) ' Samples from the first ', N, ' integers'
      WRITE (NOUT,*)
      WRITE (NOUT,*) ' Sample size     Values'
      DO 20 I = 1, N
         IPOP(I) = I
   20 CONTINUE
      DO 40 M = 1, N
         IFAIL = 0
         CALL G05NBF(IPOP,N,ISAMPL,M,IGEN,ISEED,IFAIL)
*
         WRITE (NOUT,99998) M, (ISAMPL(K),K=1,M)
   40 CONTINUE
      STOP
*
99999 FORMAT (1X,A,I1,A)
99998 FORMAT (1X,I6,10X,8I3)
      END
```

9.2 Program Data

None.

9.3 Program Results

```
 G05NBF Example Program Results

  Samples from the first 8 integers

  Sample size     Values
      1            3
      2            2  5
      3            3  4  8
      4            2  4  5  6
      5            1  2  4  6  8
      6            1  2  3  5  7  8
      7            1  2  3  4  5  6  8
      8            1  2  3  4  5  6  7  8
```