# NAG Fortran Library Routine Document

# G05QDF

**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

G05QDF generates a random two-way table.

## 2 Specification

```
 SUBROUTINE G05QDF(MODE, NROW, NCOL, TOTR, TOTC, X, LDX, IGEN, ISEED, R,
1                  NR, IW, LIW, IFAIL)
 INTEGER           MODE, NROW, NCOL, TOTR(NROW), TOTC(NCOL), X(LDX,NCOL),
1                  LDX, IGEN, ISEED(4), NR, IW(LIW), LIW, IFAIL
 real              R(NR)
```

## 3 Description

Given $m$ row totals $R_i$ and $n$ column totals $C_j$ (with $\sum_{i=1}^{m} R_i = \sum_{j=1}^{n} C_j = T$, say), G05QDF will generate a pseudorandom two-way table of integers such that the row and column totals are satisfied.

The method used is based on that described by Patefield (1981) which is most efficient when $T$ is large relative to the number of table entries $m \times n$ (i.e., $T > 2mn$). Entries are generated one row at a time and one entry at a time within a row. Each entry is generated using the conditional probability distribution for that entry given the entries in the previous rows and the previous entries in the same row.

A reference vector is used to store computed values that can be reused in the generation of new tables with the same row and column totals. G05QDF can be called to simply set up the reference vector, or to generate a two-way table using a reference vector set up in a previous call, or it can combine both functions in a single call.

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 G05QDF.

## 4 References

Patefield WM (1981) An efficient method of generating $R \times C$ tables with given row and column totals *Appl. Stats.* **30** 91–97

## 5 Parameters

1: MODE – INTEGER *Input*

*On entry*: a code for selecting the operation to be performed by the routine:

MODE $= 0$

Set up reference vector only.

MODE $= 1$

Generate two-way table using reference vector set up in a prior call to G05QDF.

MODE $= 2$

Set up reference vector and generate two-way table.

*Constraint*: MODE $= 0$, $1$ or $2$.

2: NROW – INTEGER *Input*

*On entry*: the number of rows in the table, $m$.

*Constraint*: NROW $\geq 2$.

3: NCOL – INTEGER *Input*

*On entry*: the number of columns in the table, $n$.

*Constraint*: NCOL $\geq 2$.

4: TOTR(NROW) – INTEGER array *Input*

*On entry*: the $m$ row totals, $R_i$, for $i = 1, 2, \ldots, m$.

*Constraints*:

TOTR$(i) \geq 0$, for $i = 1, 2, \ldots, m$, and $\sum_{i=1}^{m}$TOTR$(i) = \sum_{j=1}^{n}$TOTC$(j)$.

5: TOTC(NCOL) – INTEGER array *Input*

*On entry*: the $n$ column totals, $C_j$, for $j = 1, 2, \ldots, n$.

*Constraints*:

TOTC$(j) \geq 0$, for $j = 1, 2, \ldots, n$, and $\sum_{j=1}^{n}$TOTC$(j) = \sum_{i=1}^{m}$TOTR$(i)$.

6: X(LDX,NCOL) – INTEGER array *Output*

*On exit*: a pseudo-random two-way $m$ by $n$ table, $X$, with element X$(i, j)$ containing the $(i, j)$th entry in the table such that $\sum_{i=1}^{\text{NROW}}$ X$(i, j) =$ TOTC$(j)$ and $\sum_{j=1}^{\text{NCOL}}$ X$(i, j) =$ TOTR$(i)$

7: LDX – INTEGER *Input*

*On entry*: the first dimension of the array X as declared in the (sub)program from which G05QDF is called.

*Constraint*: LDX $\geq$ NROW.

8: 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.

9: 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.

10: R(NR) – ***real*** array *Input/Output*

*On exit*: the reference vector.

11: NR – INTEGER *Input*

*On entry*: the dimension of the array R as declared in the (sub)program from which G05QDF is called.

*Constraint*: NR $\geq \sum_{i=1}^{\text{NROW}}$ RTOT$(i) + 4$.

12: IW(LIW) – INTEGER array *Workspace*
13: LIW – INTEGER *Input*

*On entry*: the first dimension of the array IW as declared in the (sub)program from which G05QDF is called.

*Constraint*: LIW $\geq$ NCOL.

14: 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, MODE $< 0$,
or MODE $> 2$.

IFAIL $= 2$

On entry, NROW $< 2$,
or NCOL $< 2$.

IFAIL $= 3$

On entry, LDX $<$ NROW.

IFAIL $= 4$

At least one element of TOTR or TOTC is negative.

IFAIL $= 5$

The arrays TOTR and TOTC do not sum to the same total.

IFAIL $= 6$

On entry, NR is too small when MODE $= 0$ or 2 (see Section 5).

IFAIL $= 7$

The value of NROW or NCOL is not the same as when R was set up in a previous call with MODE $= 0$ or 2.

IFAIL $= 8$

On entry, LIW $<$ NCOL.

## 7 Accuracy

None.

# 8 Further Comments

None.

# 9 Example

Following initialisation of the pseudo random number generator by a call to G05KBF, a 4 by 3 two-way table, with row totals of 9, 11, 7 and 23 respectively, and column totals of 16, 17 and 17 respectively, is generated and printed.

## 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.

```
*     G05QDF Example Program Text
*     Mark 20 Release. NAG Copyright 2001.
*     .. Parameters ..
      INTEGER          NROW, NCOL, LDX, NOUT, NIR, NR
      PARAMETER        (NROW=4,NCOL=3,LDX=NROW,NOUT=6,NIR=10,NR=60)
*     .. Local Scalars ..
      INTEGER          I, IFAIL, IGEN, J, RCTOT
*     .. Local Arrays ..
      real             R(NR)
      INTEGER          IR(NIR), ISEED(4), TOTC(NCOL), TOTR(NROW),
     +                 X(LDX,NCOL)
*     .. External Subroutines ..
      EXTERNAL         G05KBF, G05QDF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G05QDF Example Program Results'
      WRITE (NOUT,*)
*     Set the table row and column totals
      TOTR(1) = 9
      TOTR(2) = 11
      TOTR(3) = 7
      TOTR(4) = 23
      TOTC(1) = 16
      TOTC(2) = 17
      TOTC(3) = 17
      RCTOT = 50
*     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)
*     Choose MODE = 2
      IFAIL = 0
      CALL G05QDF(2,NROW,NCOL,TOTR,TOTC,X,LDX,IGEN,ISEED,R,NR,IR,NIR,
     +            IFAIL)
*
      DO 20 I = 1, NROW
         WRITE (NOUT,99999) (X(I,J),J=1,NCOL), TOTR(I)
   20 CONTINUE
      WRITE (NOUT,*) ' --------------+-------'
      WRITE (NOUT,99999) (TOTC(J),J=1,NCOL), RCTOT
      STOP
*
99999 FORMAT (1X,3(I4,1X),'|',1X,I5)
      END
```

## 9.2 Program Data

None.

## 9.3 Program Results

```
 G05QDF Example Program Results

    3     1     5 |      9
    4     3     4 |     11
    0     5     2 |      7
    9     8     6 |     23
  --------------+-------
   16    17    17 |     50
```