# NAG Library Routine Document

# G05PZF

**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

G05PZF generates a random two-way table.

## 2 Specification

```
 SUBROUTINE G05PZF(MODE, NROW, NCOL, TOTR, TOTC, R, LR, STATE, X, LDX,
1                  IFAIL)
 INTEGER           MODE, NROW, NCOL, TOTR(NROW), TOTC(NCOL), LR,
1                  STATE(lstate), X(LDX,NCOL), LDX, IFAIL
 double precision  R(LR)
```

## 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), G05PZF 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. G05PZF 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 initialization routines G05KFF (for a repeatable sequence if computed sequentially) or G05KGF (for a non-repeatable sequence) must be called prior to the first call to G05PZF.

## 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 G05PZF.

MODE $= 2$
: Set up reference vector and generate two-way table.

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

2: NROW – INTEGER *Input*

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

*Constraint*: NROW $\geq 2$.

3: NCOL – INTEGER *Input*

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

*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*:

$$\begin{aligned}&\text{TOTR}(i) \geq 0, \text{ for } i = 1, 2, \ldots, m;\\ &\sum_{i=1}^{m}\text{TOTR}(i) = \sum_{j=1}^{n}\text{TOTC}(j);\\ &\textstyle\sum_i \text{TOTR}(i) > 0, \text{ for } i = 1, 2, \ldots, \text{NROW}.\end{aligned}$$

5: TOTC(NCOL) – INTEGER array *Input*

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

*Constraints*:

$$\begin{aligned}&\text{TOTC}(j) \geq 0, \text{ for } j = 1, 2, \ldots, n;\\ &\sum_{j=1}^{n}\text{TOTC}(j) = \sum_{i=1}^{m}\text{TOTR}(i).\end{aligned}$$

6: R(LR) – ***double precision*** array *Communication Array*

*On entry*: if MODE $= 1$, the reference vector from the previous call to G05PZF.

*On exit*: the reference vector.

7: LR – INTEGER *Input*

*On entry*: the dimension of the array R as declared in the (sub)program from which G05PZF is called.

*Constraint*: LR $\geq \displaystyle\sum_{i=1}^{m}\text{TOTR}(i) + 5$.

8: STATE($lstate$) – INTEGER array *Communication Array*

**Note**: $lstate$ = LSTATE, the dimension of STATE as supplied to the initialization routine G05KFF or G05KGF.

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

*On exit*: contains updated information on the state of the generator.

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

*On exit*: if MODE $= 1$ or 2, a pseudorandom two-way $m$ by $n$ table, $X$, with element X$(i, j)$ containing the $(i, j)$th entry in the table such that $\displaystyle\sum_{i=1}^{m}\text{X}(i, j) = \text{TOTC}(j)$ and $\displaystyle\sum_{j=1}^{n}\text{X}(i, j) = \text{TOTR}(i)$

10: LDX – INTEGER *Input*

*On entry*: the first dimension of the array X as declared in the (sub)program from which G05PZF is called.

*Constraint*: LDX $\geq$ NROW.

11: 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, MODE $\neq 0$, 1 or 2.

IFAIL $= 2$

On entry, NROW $< 2$.

IFAIL $= 3$

On entry, NCOL $< 2$.

IFAIL $= 4$

At least one element of TOTR is negative.

IFAIL $= 5$

At least one element of TOTC is negative.

IFAIL $= 6$

The value of NROW or NCOL is not the same as when R was set up in a previous call to G05PZF with MODE $= 0$ or 2.

IFAIL $= 7$

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

IFAIL $= 8$

On entry, STATE vector was not initialized or has been corrupted.

IFAIL $= 10$

On entry, LDX $<$ NROW.

IFAIL = 15

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

## 7 Accuracy

None.

## 8 Further Comments

None.

## 9 Example

Following initialization of the pseudorandom number generator by a call to G05KFF, this example generates and prints 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.

### 9.1 Program Text

```
*     G05PZF Example Program Text
*     Mark 22 Revised. NAG Copyright 2008.
*     .. Parameters ..
      INTEGER          NOUT
      PARAMETER        (NOUT=6)
      INTEGER          LROW, NCOL, LDX, MSEED, MSTATE, LR
      PARAMETER        (LROW=4,NCOL=3,LDX=LROW,MSEED=1,MSTATE=633,LR=60)
*     .. Local Scalars ..
      INTEGER          GENID, I, IFAIL, J, LSEED, LSTATE, RCTOT, SUBID
*     .. Local Arrays ..
      DOUBLE PRECISION R(LR)
      INTEGER          SEED(MSEED), STATE(MSTATE), TOTC(NCOL),
     +                 TOTR(LROW), X(LDX,NCOL)
*     .. External Subroutines ..
      EXTERNAL         G05KFF, G05PZF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G05PZF 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
*     Initialize the seed
      SEED(1) = 1762543
*     GENID and SUBID identify the base generator
      GENID = 1
      SUBID = 1
*     Initialize the generator to a repeatable sequence
      LSTATE = MSTATE
      LSEED = MSEED
      IFAIL = 1
      CALL G05KFF(GENID,SUBID,SEED,LSEED,STATE,LSTATE,IFAIL)
      IF (IFAIL.NE.0) THEN
         WRITE (NOUT,99998) IFAIL
         GO TO 40
      END IF
*     Generate the random table
*     Choose MODE = 2
      IFAIL = 1
      CALL G05PZF(2,LROW,NCOL,TOTR,TOTC,R,LR,STATE,X,LDX,IFAIL)
      IF (IFAIL.NE.0) THEN
```

```
         WRITE (NOUT,99997) IFAIL
         GO TO 40
      END IF
*     Display the results
      DO 20 I = 1, LROW
         WRITE (NOUT,99999) (X(I,J),J=1,NCOL), TOTR(I)
   20 CONTINUE
      WRITE (NOUT,*) ' --------------+-------'
      WRITE (NOUT,99999) (TOTC(J),J=1,NCOL), RCTOT
*
   40 CONTINUE
*
99999 FORMAT (1X,3(I4,1X),'|',1X,I5)
99998 FORMAT (1X,' ** G05KFF returned with IFAIL = ',I5)
99997 FORMAT (1X,' ** G05PZF returned with IFAIL = ',I5)
      END
```

## 9.2 Program Data

None.

## 9.3 Program Results

```
 G05PZF Example Program Results

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