

F01YEFP

NAG Parallel Library Routine Document

Note: before using this routine, please read the Users' Note for your implementation to check for implementation-dependent details. You are advised to enclose any calls to NAG Parallel Library routines between calls to Z01AAFP and Z01ABFP.

1 Description

F01YEFP generates in-place a real dense vector x of length n distributed on the Library Grid conformally to a sparse matrix A (see Section 2.5 of the F11 Chapter Introduction).

This routine generates distributed vectors in the form required by a number of routines in Chapter F11.

A user-supplied subroutine is required to generate a contiguous portion of the vector x .

An appropriate Chapter F11 routine must have been called prior to F01YEFP in order to set up auxiliary information about the sparse matrix A in the array IAINFO. See Section 3.3 of the F11 Chapter Introduction for further information, particularly Section 3.3.2.

2 Specification

```

SUBROUTINE F01YEFP(ICNTXT, GVEC, N, X, IAINFO, IFAIL)
DOUBLE PRECISION  X(*)
INTEGER           ICNTXT, N, IAINFO(*), IFAIL
EXTERNAL         GVEC

```

3 Usage

3.1 Definitions

The following definitions are used in describing the data distribution within this document:

m_l – the number of vector elements to be stored locally ($m_l = \text{IAINFO}(3)$, see IAINFO).

3.2 Global and Local Arguments

The following global **input** arguments must have the same value on entry to the routine on each processor and the global **output** arguments will have the same value on exit from the routine on each processor:

Global input arguments: N, IFAIL

Global output arguments: IFAIL

The remaining arguments are local.

3.3 Distribution Strategy

On exit from F01YEFP, the vector x is distributed conformally to the sparse matrix A , i.e., the vector x is distributed across the logical processors in the Library Grid in the same way as each of the columns of the matrix A . The local part of the vector x is stored in the array X. This data distribution is described in more detail in Section 2.5 of the F11 Chapter Introduction.

3.4 Related Routines

The vectors generated by F01YEFP can be gathered using the Library routine F01XFFP.

3.5 Requisites

The sparse matrix A must have been preprocessed to set up the auxiliary array IAINFO by an appropriate Chapter F11 routine.

Cyclic row block distribution: F11ZBFP or F11ZFPF for real or complex A , respectively.

4 Arguments

- 1: ICNTXT — INTEGER *Local Input*
On entry: the Library context, usually returned by a call to the Library Grid initialisation routine Z01AAFP.

Note: the value of ICNTXT **must not** be changed.

- 2: GVEC — SUBROUTINE, supplied by the user. *External Procedure*
 GVEC must return in XL the contiguous portion $x(i_1 : i_2)$ of the vector x .
 Its specification is:

SUBROUTINE	GVEC(I1, I2, XL)	
DOUBLE PRECISION	XL(*)	
INTEGER	I1, I2	
1: I1 — INTEGER		<i>Local Input</i>
<i>On entry:</i> i_1 , the index of the first element of the contiguous portion of the vector x to be generated.		
2: I2 — INTEGER		<i>Local Input</i>
<i>On entry:</i> i_2 , the index of the last element of the contiguous portion of the vector x to be generated.		
3: XL(*) — DOUBLE PRECISION array		<i>Local Output</i>
<i>On exit:</i> XL must return the contiguous portion $x(i_1 : i_2)$ of the vector x .		

GVEC must be declared as EXTERNAL in the (sub)program from which F01YEFP is called. Arguments denoted as *Input* must **not** be changed by this procedure.

- 3: N — INTEGER *Global Input*
On entry: n , the order of the vector x . It must contain the same value as the parameter N used in a prior call to the Chapter F11 routine in which the array IAINFO was initialised.

Constraint: $N \geq 1$.

- 4: X(*) — DOUBLE PRECISION array *Local Output*
Note: the dimension of the array X must be at least $\max(1, m_l)$.
On exit: the local part of the vector x .

- 5: IAINFO(*) — INTEGER array *Local Input*
Note: the dimension of the array IAINFO must be at least $\max(30, \text{IAINFO}(2))$.
On entry: the first IAINFO(2) elements of IAINFO contain information about the matrix A . The array IAINFO must have been initialised by a prior call to an appropriate Chapter F11 routine. The first IAINFO(2) elements of IAINFO **must not** be changed between successive calls to library routines involving the sparse matrix A .

Note: on exit from the Chapter F11 routine, the element IAINFO(3) contains m_l , the number of rows of the matrix stored locally, and IAINFO(5) contains m_l^{\max} , the maximum number of vector elements stored on any processor of the Library Grid.

6: IFAIL — INTEGER*Global Input/Global Output*

The NAG Parallel Library provides a mechanism, via the routine Z02EAFP, to reduce the amount of parameter validation performed by this routine. For a full description refer to the Z02 Chapter Introduction.

On entry: IFAIL must be set to 0, -1 or 1. For users not familiar with this argument (described in the Essential Introduction) the recommended values are:

IFAIL = 0, if multigridding is **not** employed;
 IFAIL = -1, if multigridding is employed.

On exit: IFAIL = 0 (or -9999 if reduced error checking is enabled) unless the routine detects an error (see Section 5).

5 Errors and Warnings

If on entry IFAIL = 0 or -1, explanatory error messages are output from the root processor (or processor {0,0} when the root processor is not available) on the current error message unit (as defined by X04AAF).

5.1 Full Error Checking Mode Only

IFAIL = -2000

The routine has been called with an invalid value of ICNTXT on one or more processors.

IFAIL = -1000

The logical processor grid and library mechanism (Library Grid) have not been correctly defined, see Z01AAFP.

IFAIL = -*i*

On entry, the *i*th argument was invalid. This error occurred either because a global argument did not have the same value on all logical processors, or because its value on one or more processors was incorrect. An explanatory message distinguishes between these two cases.

5.2 Any Error Checking Mode

IFAIL = 1

IAINFO was not initialised by a prior call to a Chapter F11 routine.

IFAIL = 2

On entry, the data stored in the arguments N and IAINFO are inconsistent. This could indicate that, after the array IAINFO was initialised, at least one of the arguments was changed between successive calls to library routines.

6 Further Comments

6.1 Algorithmic Detail

This routine successively calls the user-supplied generation routine for each contiguous portion of the vector *x* to be stored locally.

6.2 Parallelism Detail

This routine generates the local portions of the vector *x* independently on each logical processor.

7 References

None.

8 Example

See Section 8 of the document for F11DHFP.
