

E04MGF – NAG Fortran Library Routine Document

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

To supply optional parameters to E04MFF from an external file.

2 Specification

```
SUBROUTINE E04MGF(IOPTNS, INFORM)
INTEGER          IOPTNS, INFORM
```

3 Description

E04MGF may be used to supply values for optional parameters to E04MFF. E04MGF reads an external file and each line of the file defines a single optional parameter. It is only necessary to supply values for those parameters whose values are to be different from their default values.

Each optional parameter is defined by a single character string of up to 72 characters, consisting of one or more items. The items associated with a given option must be separated by spaces, or equal signs [=]. Alphabetic characters may be upper or lower case. The string

```
Print level = 1
```

is an example of a string used to set an optional parameter. For each option the string contains one or more of the following items:

- (a) A mandatory keyword.
- (b) A phrase that qualifies the keyword.
- (c) A number that specifies an INTEGER or *real* value. Such numbers may be up to 16 contiguous characters in Fortran 77's I, F, E or D formats, terminated by a space if this is not the last item on the line.

Blank strings and comments are ignored. A comment begins with an asterisk (*) and all subsequent characters in the string are regarded as part of the comment.

The file containing the options must start with **begin** and must finish with **end**. An example of a valid options file is:

```
Begin * Example options file
Print level = 10
End
```

Normally each line of the file is printed as it is read, on the current advisory message unit (see X04ABF), but printing may be suppressed using the keyword **nolist**. To suppress printing of **begin**, **nolist** must be the first option supplied as in the file:

```
Begin
Nolist
Print level = 10
End
```

Printing will automatically be turned on again after a call to E04MFF and may be turned on again at any time by the user by using the keyword **list**.

Optional parameter settings are preserved following a call to E04MFF, and so the keyword **defaults** is provided to allow the user to reset all the optional parameters to their default values prior to a subsequent call to E04MFF.

A complete list of optional parameters, their abbreviations, synonyms and default values is given in Section 11 of the document for E04MFF.

4 References

None.

5 Parameters

1: IOPTNS — INTEGER *Input*

On entry: the unit number of the options file to be read.

Constraint: $0 \leq \text{IOPTNS} \leq 99$.

2: INFORM — INTEGER *Output*

On exit: contains zero if the options file has been successfully read and a value > 0 otherwise, as indicated below.

INFORM = 1

IOPTNS is not in the range [0, 99].

INFORM = 2

begin was found, but end-of-file was found before **end** was found.

INFORM = 3

end-of-file was found before **begin** was found.

6 Error Indicators and Warnings

If a line is not recognized as a valid option, then a warning message is output on the current advisory message unit (see X04ABF).

7 Accuracy

Not applicable.

8 Further Comments

E04MHF may also be used to supply optional parameters to E04MFF.

9 Example

This example solves the same problem as the example for E04MFF, but in addition illustrates the use of E04MGF and E04MHF to set optional parameters for E04MFF.

In this example the options file read by E04MGF is appended to the data file for the program (see Section 9.2). It would usually be more convenient in practice to keep the data file and the options file separate.

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.

```
*      E04MGF Example Program Text
*      Mark 18 Revised.  NAG Copyright 1997.
*      .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
```

```

      INTEGER          NMAX, NCMAX
      PARAMETER       (NMAX=10,NCMAX=10)
      INTEGER          LDA
      PARAMETER       (LDA=NCMAX)
      INTEGER          LIWORK, LWORK
      PARAMETER       (LIWORK=1000,LWORK=10000)
*   .. Local Scalars ..
      real            OBJ
      INTEGER          I, IFAIL, INFORM, ITER, J, N, NCLIN
*   .. Local Arrays ..
      real            A(LDA,NMAX), AX(NCMAX), BL(NMAX+NCMAX),
+                   BU(NMAX+NCMAX), CLAMDA(NMAX+NCMAX), CVEC(NMAX),
+                   WORK(LWORK), X(NMAX)
      INTEGER          ISTATE(NMAX+NCMAX), IWORK(LIWORK)
*   .. External Subroutines ..
      EXTERNAL        E04MFF, E04MGF, E04MHF, X04ABF
*   .. Executable Statements ..
      WRITE (NOUT,*) 'E04MGF Example Program Results'
*   Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N, NCLIN
      IF (N.LE.NMAX .AND. NCLIN.LE.NCMAX) THEN
*
*       Read CVEC, A, BL, BU and X from data file
*
      READ (NIN,*) (CVEC(I),I=1,N)
      READ (NIN,*) ((A(I,J),J=1,N),I=1,NCLIN)
      READ (NIN,*) (BL(I),I=1,N+NCLIN)
      READ (NIN,*) (BU(I),I=1,N+NCLIN)
      READ (NIN,*) (X(I),I=1,N)
*
*       Set three options using E04MHF
*
      CALL E04MHF(' Print Level = 1 ')
*
      CALL E04MHF(' Check Frequency = 10 ')
*
      CALL E04MHF(' Infinite Bound Size = 1.0D+25 ')
*
*       Set the unit number for advisory messages to NOUT
*
      CALL X04ABF(1,NOUT)
*
*       Read the options file for the remaining options
*
      CALL E04MGF(NIN,INFORM)
*
      IF (INFORM.NE.0) THEN
+         WRITE (NOUT,99999) 'E04MGF terminated with INFORM = ',
+             INFORM
          STOP
      END IF
*

```

```

*       Solve the problem
*
*       IFAIL = -1
*
*       CALL E04MFF(N,NCLIN,A,LDA,BL,BU,CVEC,ISTATE,X,ITER,OBJ,AX,
+           CLAMDA,IWORK,LIWORK,WORK,LWORK,IFAIL)
*
*       END IF
*       STOP
*
* 99999 FORMAT (1X,A,I3)
*       END

```

9.2 Program Data

E04MGF Example Program Data

```

7 7                                     :Values of N and NCLIN
-0.02 -0.20 -0.20 -0.20 -0.20  0.04  0.04 :End of CVEC
 1.00  1.00  1.00  1.00  1.00  1.00  1.00
 0.15  0.04  0.02  0.04  0.02  0.01  0.03
 0.03  0.05  0.08  0.02  0.06  0.01  0.00
 0.02  0.04  0.01  0.02  0.02  0.00  0.00
 0.02  0.03  0.00  0.00  0.01  0.00  0.00
 0.70  0.75  0.80  0.75  0.80  0.97  0.00
 0.02  0.06  0.08  0.12  0.02  0.01  0.97 :End of matrix A
-0.01 -0.10 -0.01 -0.04 -0.10 -0.01 -0.01
-0.13 -1.0E+25 -1.0E+25 -1.0E+25 -1.0E+25 -9.92E-02 -3.0E-03 :End of BL
 0.01  0.15  0.03  0.02  0.05  1.0E+25  1.0E+25
-0.13 -4.9E-03 -6.4E-03 -3.7E-03 -1.2E-03  1.0E+25  2.0E-03 :End of BU
-0.01 -0.03  0.00 -0.01 -0.10  0.02  0.01 :End of X
Begin Example options file for E04MGF
  Crash Tolerance = 0.05 * (Default = 0.01)
  Iteration Limit = 25 * (Default = 70)
End

```

9.3 Program Results

E04MGF Example Program Results

Calls to E04MHF

```

Print Level = 1
Check Frequency = 10
Infinite Bound Size = 1.0E+25

```

OPTIONS file

```

Begin Example options file for E04MGF
  Crash Tolerance = 0.05 * (Default = 0.01)
  Iteration Limit = 25 * (Default = 70)
End

```

*** E04MFF
 *** Start of NAG Library implementation details ***

Implementation title: Generalised Base Version
 Precision: FORTRAN double precision
 Product Code: FLBAS19D
 Mark: 19A

*** End of NAG Library implementation details ***

Parameters

Problem type..... LP

Linear constraints..... 7 Feasibility tolerance.. 1.05E-08
 Variables..... 7 Optimality tolerance... 1.72E-13

Infinite bound size.... 1.00E+25 COLD start.....
 Infinite step size.... 1.00E+25 EPS (machine precision) 1.11E-16

Check frequency..... 10 Expand frequency..... 5
 Minimum sum of infeas.. NO Crash tolerance..... 5.00E-02

Print level..... 1 Iteration limit..... 25
 Monitoring file..... -1

Workspace provided is IWORK(1000), WORK(10000).
 To solve problem we need IWORK(17), WORK(182).

Varbl	State	Value	Lower Bound	Upper Bound	Lagr Mult	Slack
V 1	LL	-1.000000E-02	-1.000000E-02	1.000000E-02	0.3301	.
V 2	LL	-0.100000	-0.100000	0.150000	1.4384E-02	.
V 3	UL	3.000000E-02	-1.000000E-02	3.000000E-02	-9.0997E-02	.
V 4	UL	2.000000E-02	-4.000000E-02	2.000000E-02	-7.6612E-02	.
V 5	FR	-6.748534E-02	-0.100000	5.000000E-02	.	3.2515E-02
V 6	FR	-2.280130E-03	-1.000000E-02	None	.	7.7199E-03
V 7	FR	-2.345277E-04	-1.000000E-02	None	.	9.7655E-03

L Con	State	Value	Lower Bound	Upper Bound	Lagr Mult	Slack
L 1	EQ	-0.130000	-0.130000	-0.130000	-1.431	2.7756E-17
L 2	FR	-5.479544E-03	None	-4.900000E-03	.	5.7954E-04
L 3	FR	-6.571922E-03	None	-6.400000E-03	.	1.7192E-04
L 4	FR	-4.849707E-03	None	-3.700000E-03	.	1.1497E-03
L 5	FR	-3.874853E-03	None	-1.200000E-03	.	2.6749E-03
L 6	LL	-9.920000E-02	-9.920000E-02	None	1.501	-1.3878E-17
L 7	LL	-3.000000E-03	-3.000000E-03	2.000000E-03	1.517	.

Exit E04MFF - Optimal LP solution.

Final LP objective value = 0.2359648E-01

Exit from LP problem after 5 iterations.