

## E04DJF – 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 E04DGF from an external file.

### 2 Specification

```
SUBROUTINE E04DJF(IOPTNS, INFORM)
  INTEGER          IOPTNS, INFORM
```

### 3 Description

E04DJF may be used to supply values for optional parameters to E04DGF. E04DJF 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 = 5
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 = 5
End
```

Printing will automatically be turned on again after a call to E04DGF 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 E04DGF, 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 E04DGF.

A complete list of optional parameters, their abbreviations, synonyms and default values is given in Section 11 of the document for E04DGF.

## 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 an options file with the correct structure has been 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

E04DKF may also be used to supply optional parameters to E04DGF.

## 9 Example

This example solves the same problem as the example for E04DGF, but in addition illustrates the use of E04DJF and E04DKF to set optional parameters for E04DGF.

In this example the options file read by E04DJF is appended to the data file for the program (see Section 9.1). 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.

```
*      E04DJF Example Program Text
*      Mark 16 Release. NAG Copyright 1993.
*      .. Parameters ..
      INTEGER          NMAX
      PARAMETER       (NMAX=10)
```

```

      INTEGER          NIN, NOUT
      PARAMETER       (NIN=5,NOUT=6)
*   .. Local Scalars ..
      real           OBJF
      INTEGER         I, IFAIL, INFORM, ITER, N
*   .. Local Arrays ..
      real           OBJGRD(NMAX), USER(4*NMAX), WORK(13*NMAX),
+                   X(NMAX)
      INTEGER         IUSER(NMAX), IWORK(NMAX+1)
*   .. External Subroutines ..
      EXTERNAL        E04DGF, E04DJF, E04DKF, OBJFN1, X04ABF
*   .. Executable Statements ..
      WRITE (NOUT,*) 'E04DJF Example Program Results'
*   Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) N
      IF (N.LE.NMAX) THEN
*
*       Read X from data file
*
      READ (NIN,*) (X(I),I=1,N)
*
*       Set two options using E04DKF
*
      CALL E04DKF(' Verify Level = -1 ')
*
      CALL E04DKF(' Maximum Step Length = 100.0 ')
*
*       Set the unit number for advisory messages to NOUT
*
      CALL X04ABF(1,NOUT)
*
*       Read the options file for the remaining options
*
      CALL E04DJF(NIN,INFORM)
*
      IF (INFORM.NE.0) THEN
+         WRITE (NOUT,99999) 'E04DJF terminated with INFORM = ',
          INFORM
          STOP
      END IF
*
*       Solve the problem
*
      IFAIL = -1
*
      CALL E04DGF(N,OBJFN1,ITER,OBJF,OBJGRD,X,IWORK,WORK,IUSER,USER,
+              IFAIL)
*
      END IF
      STOP
*
99999 FORMAT (1X,A,I3)
      END
*

```

```

SUBROUTINE OBJFN1(MODE,N,X,OBJF,OBJGRD,NSTATE,IUSER,USER)
*   Routine to evaluate F(x) and approximate its 1st derivatives
*   .. Scalar Arguments ..
  real          OBJF
  INTEGER       MODE, N, NSTATE
*   .. Array Arguments ..
  real          OBJGRD(N), USER(*), X(N)
  INTEGER       IUSER(*)
*   .. Local Scalars ..
  real          EPSRF
  INTEGER       I, IFAIL, IMODE, IWARN, LHES, MSGLVL
*   .. Local Arrays ..
  real          USE(1)
  INTEGER       IUSE(1)
*   .. External Subroutines ..
  EXTERNAL     E04XAF, OBJFN2
*   .. Executable Statements ..
  IF (MODE.EQ.0) THEN
    Evaluate F(x) only
    CALL OBJFN2(MODE,N,X,OBJF,OBJGRD,NSTATE,IUSE,USE)
*
  ELSE IF (MODE.EQ.2) THEN
    Evaluate F(x) and approximate its 1st derivatives
    MSGLVL = 0
    EPSRF = 0.0e0
    IMODE = 0
    LHES = N
    DO 20 I = 1, N
      USER(I) = 0.0e0
20  CONTINUE
    IFAIL = 1
*
    CALL E04XAF(MSGLVL,N,EPSRF,X,IMODE,OBJFN2,LHES,USER(1),OBJF,
+             OBJGRD,USER(N+1),USER(2*N+1),IWARN,USER(3*N+1),
+             IUSE,USE,IUSER,IFAIL)
*
  END IF
*
  RETURN
END
SUBROUTINE OBJFN2(MODE,N,X,OBJF,OBJGRD,NSTATE,IUSE,USE)
*   Routine to evaluate F(x)
*   .. Scalar Arguments ..
  real          OBJF
  INTEGER       MODE, N, NSTATE
*   .. Array Arguments ..
  real          OBJGRD(N), USE(*), X(N)
  INTEGER       IUSE(*)
*   .. Local Scalars ..
  real          X1, X2
*   .. Intrinsic Functions ..
  INTRINSIC     EXP
*   .. Executable Statements ..
  X1 = X(1)
  X2 = X(2)
*
  OBJF = EXP(X1)*(4.0e0*X1**2+2.0e0*X2**2+4.0e0*X1*X2+2.0e0*X2+

```

```

+      1.0e0)
*
  RETURN
  END

```

### 9.2 Program Data

```

E04DJF Example Program Data
  2                               :Value of N
-1.0 1.0                         :End of X
Begin Example options file for E04DJF
  Iteration Limit = 25           * (Default = 50)
  Print Level     = 1            * (Default = 10)
End

```

### 9.3 Program Results

E04DJF Example Program Results

Calls to E04DKF  
-----

Verify Level = -1  
Maximum Step Length = 100.0

OPTIONS file  
-----

```

Begin Example options file for E04DJF
  Iteration Limit = 25           * (Default = 50)
  Print Level     = 1            * (Default = 10)
End

```

\*\*\* E04DGF  
\*\*\* Start of NAG Library implementation details \*\*\*

Implementation title: Generalised Base Version  
Precision: FORTRAN double precision  
Product Code: FLBAS18D  
Mark: 17A

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

Parameters  
-----

```

Variables..... 2

Maximum step length... 1.00E+02   EPS (machine precision) 1.11E-16
Optimality tolerance... 3.26E-12   Linesearch tolerance... 9.00E-01

Est. opt. function val.   None     Function precision..... 4.37E-15
Verify level..... -1

Iteration limit..... 25     Print level..... 1

```

Exit from E04DGF after 10 iterations.

Variable		Value	Gradient value
Varbl	1	0.500000	9.1E-07
Varbl	2	-1.000000	8.3E-07

Exit E04DGF - Optimal solution found.

Final objective value = 0.5308300E-13

---