

NAG Toolbox for MATLAB

e04nm

1 Purpose

To supply individual optional parameters to e04nk

2 Syntax

```
[lwsav, iwsav, rwsav, inform] = e04nm(str, lwsav, iwsav, rwsav)
```

3 Description

e04nm may be used to supply values for optional parameters to e04nk. It is only necessary to call e04nm for those parameters whose values are to be different from their default values. One call to e04nm sets one parameter value.

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 equals 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 mandatory keyword;
- a phrase that qualifies the keyword;
- a number that specifies an integer or double value. Such numbers may be up to 16 contiguous characters in Fortran'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.

For e04nm, each user-specified option is normally printed as it is defined, on the current advisory message unit (see x04ab), but this printing may be suppressed using the keyword **Nolist**. Thus the statement

```
[lwsav, iwsav, rwsav, inform] = e04nm('Nolist', lwsav, iwsav, rwsav);
```

suppresses printing of this and subsequent options. Printing will automatically be turned on again after a call to e04nk and may be turned on again at any time using the keyword **List**.

For e04nm printing is turned off by default, but may be turned on at any time using the keyword **List**.

Optional parameter settings are preserved following a call to e04nk and so the keyword **Defaults** is provided to allow you to reset all the optional parameters to their default values before a subsequent call to e04nk.

A complete list of optional parameters, their abbreviations, synonyms and default values is given in Section 11 of the document for e04nk.

4 References

None.

5 Parameters

5.1 Compulsory Input Parameters

1: **str** – string

A single valid option string (as described in Section 3 and in Section 11 of the document for e04nk).

2: **lwsav(20)** – logical array

3: **iwsav(380)** – int32 array

4: **rwsav(285)** – double array

The arrays **lwsav**, **iwsav** and **rwsav** **must not** be altered between calls to any of the functions e04wb, e04nk or e04nm.

5.2 Optional Input Parameters

None.

5.3 Input Parameters Omitted from the MATLAB Interface

None.

5.4 Output Parameters

1: **lwsav(20)** – logical array

2: **iwsav(380)** – int32 array

3: **rwsav(285)** – double array

The arrays **lwsav**, **iwsav** and **rwsav** **must not** be altered between calls to any of the functions e04wb, e04nk or e04nm.

4: **inform** – int32 scalar

Contains zero if a valid option string has been supplied and a value > 0 otherwise (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

inform = 5

The supplied option is invalid. Check that the keywords are neither ambiguous nor misspelt.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

```
e04nk_qphx.m

function [hx, user] = qphx(nstate, ncolh, x, user)
    hx = zeros(ncolh, 1);

    hx(1) = 2*x(1);
```

```
hx(2) = 2*x(2);  
hx(3) = 2*(x(3)+x(4));  
hx(4) = hx(3);  
hx(5) = 2*x(5);  
hx(6) = 2*(x(6)+x(7));  
hx(7) = hx(6);
```

```
n = int32(7);  
m = int32(8);  
iobj = int32(8);  
ncolh = int32(7);  
a = [0.02;  
      0.02;  
      0.03;  
      1;  
      0.7;  
      0.02;  
      0.15;  
      -200;  
      0.06;  
      0.75;  
      0.03;  
      0.04;  
      0.05;  
      0.04;  
      1;  
      -2000;  
      0.02;  
      1;  
      0.01;  
      0.08;  
      0.08;  
      0.8;  
      -2000;  
      1;  
      0.12;  
      0.02;  
      0.02;  
      0.75;  
      0.04;  
      -2000;  
      0.01;  
      0.8;  
      0.02;  
      1;  
      0.02;  
      0.06;  
      0.02;  
      -2000;  
      1;  
      0.01;  
      0.01;  
      0.97;  
      0.01;  
      400;  
      0.97;  
      0.03;  
      1;  
      400];  
ha = [int32(7);  
      int32(5);  
      int32(3);  
      int32(1);  
      int32(6);  
      int32(4);  
      int32(2);  
      int32(8);  
      int32(7);
```

```

int32(6);
int32(5);
int32(4);
int32(3);
int32(2);
int32(1);
int32(8);
int32(2);
int32(1);
int32(4);
int32(3);
int32(7);
int32(6);
int32(8);
int32(1);
int32(7);
int32(3);
int32(4);
int32(6);
int32(2);
int32(8);
int32(5);
int32(6);
int32(7);
int32(1);
int32(2);
int32(3);
int32(4);
int32(8);
int32(1);
int32(2);
int32(3);
int32(6);
int32(7);
int32(8);
int32(7);
int32(2);
int32(1);
int32(8)];
ka = [int32(1);
int32(9);
int32(17);
int32(24);
int32(31);
int32(39);
int32(45);
int32(49)];
bl = [0;
0;
400;
100;
0;
0;
0;
2000;
-9.999999999999999e+24;
-9.999999999999999e+24;
-9.999999999999999e+24;
-9.999999999999999e+24;
1500;
250;
-9.999999999999999e+24];
bu = [200;
2500;
800;
700;
1500;
9.999999999999999e+24;
9.999999999999999e+24;
2000;

```

```
60;
100;
40;
30;
9.999999999999999e+24;
300;
9.999999999999999e+24];
start = 'C';
names = {''; '''; '''; '''; '''; '''; '''; '''; '''};
cname = {'...X1...'; '...X2...'; '...X3...'; '...X4...'; '...X5...';
'...X6...'; '...X7...'; '..ROW1..'; '..ROW2..'; '..ROW3..'; '..ROW4..';
'..ROW5..'; '..ROW6..'; '..ROW7..'; '..COST..' };
ns = int32(-1232765364);
xs = [0;
0;
0;
0;
0;
0;
0;
0;
0;
0;
0;
0;
0;
0;
0;
0;
0;
0;
0;
0;
0];
istate = zeros(15, 1, 'int32');
leniz = int32(10000);
lenz = int32(10000);
[cwsav, lwsav, iwsav, rwsav, ifail] = e04wb('e04nk');
[lwsav, iwsav, rwsav, inform] = e04nm('Print Level = 1', lwsav, iwsav,
rwsav);
[nsOut, xsOut, istateOut, miniz, minz, ninf, sinf, obj, clamda, user,
lwsavOut, iwsavOut, rwsavOut, ifail] = ...
    e04nk(n, m, iobj, ncolh, 'e04nk_qphx', a, ha, ka, bl, bu, start,
names, ...
    cname, ns, xs, istate, leniz, lenz, lwsav, iwsav, rwsav)
```

*** E04NKF

Parameters

Frequencies.		
Check frequency.....	60	Expand frequency.....
10000		
Factorization frequency.	100	
LP Parameters.		
Scale tolerance.....	9.00E-01	Feasibility tolerance...
1.00E-06		
Iteration limit.....	75	Scale option.....
2		
Optimality tolerance....	1.00E-06	Partial price.....
10		
Crash tolerance.....	1.00E-01	Pivot tolerance.....
2.05E-11		
Crash option.....	2	
QP objective.		
Objective variables.....	7	Hessian columns.....
7		
Superbasics limit.....	7	
Miscellaneous.		
Variables.....	7	Linear constraints.....
8		
LU factor tolerance.....	1.00E+02	LU update tolerance.....

```

1.00E+01
LU singularity tolerance 2.05E-11 Monitoring file.....
-1
EPS (machine precision). 1.11E-16 Print level.....
1
Infinite bound size..... 1.00E+20 Infinite step size.....
1.00E+20
COLD start..... MINIMIZE.....

Workspace provided is IZ( 10000), Z( 10000).
To start solving the problem we need IZ( 428), Z( 358).

Variable State Value Lower Bound Upper Bound Lagr Mult
Residual
...X1... LL 0.00000 . 200.00 2361.
.
...X2... BS 349.399 . 2500.0 -2.4771E-12
349.4
...X3... SBS 648.853 400.00 800.00 -1.3377E-12
151.1
...X4... SBS 172.847 100.00 700.00 -9.7454E-13
72.85
...X5... BS 407.521 . 1500.0 -9.1881E-13
407.5
...X6... BS 271.356 . None -5.9639E-13
271.4
...X7... BS 150.023 . None 1.0990E-12
150.0

Constrnt State Value Lower Bound Upper Bound Lagr Mult
Residual
..ROW1.. EQ 2000.00 2000.0 2000.0 -1.2901E+04
.
..ROW2.. BS 49.2316 None 60.000 . -
10.77
..ROW3.. UL 100.000 None 100.00 -2325.
.
..ROW4.. BS 32.0719 None 40.000 . -
7.928
..ROW5.. BS 14.5572 None 30.000 . -
15.44
..ROW6.. LL 1500.00 1500.0 None 1.4455E+04
.
..ROW7.. LL 250.000 250.00 300.00 1.4581E+04
.
..COST.. BS -2.988690E+06 None None -1.000 -
2.9887E+06

Exit E04NKF - Optimal QP solution found.

Final QP objective value = -1847785.

Exit from QP problem after 9 iterations.
nsOut =
2
xsOut =
1.0e+06 *
0
0.0003
0.0006
0.0002
0.0004
0.0003
0.0002
0.0020
0.0000

```

```

0.0001
0.0000
0.0000
0.0015
0.0002
-2.9887
istateOut =
    0
    3
    2
    2
    3
    3
    3
    0
    3
    1
    3
    3
    0
    0
    3
miniz =
    428
minz =
    358
ninf =
    0
sinf =
    0
obj =
-1.8478e+06
clamda =
    1.0e+04 *
    0.2361
    -0.0000
    -0.0000
    -0.0000
    -0.0000
    -0.0000
    0.0000
    -1.2901
    0
    -0.2325
    0
    0
    1.4455
    1.4581
    -0.0001
user =
    0
lwsavOut =
    0
    0
    1
    1
    1
    1
    1
    1
    1
    1
    1
    1
    1
    1
    1
    0
    1
    1
    0
    0
    0
    0
    0
    0

```

```
        0
        0
iwsavOut =
    array elided
rwsavOut =
    array elided
ifail =
        0
```
