nag_opt_sparse_mps_free (e04myc)

1. Purpose

nag_opt_sparse_mps_free (e04myc) frees the memory allocated by nag_opt_sparse_mps_read (e04mzc).

2. Specification

#include <nag.h>
#include <nage04.h>

3. Description

This function should be used in conjuction with nag_opt_sparse_mps_read (e04mzc), which reads data for a sparse linear or quadratic programming problem from an MPSX file, allocates several arrays, and initializes them with the data contained in the file. nag_opt_sparse_mps_free is a utility provided for the convenient freeing of this memory. It should be called in order to conserve memory which is no longer required, e.g., following a call to nag_opt_sparse_convex_qp (e04nkc) (which may be used to solve the problem defined by the MPSX file). Any memory not freed will, of course, be freed when the user's program terminates.

nag_opt_sparse_mps_free can be used to free a subset of the allocated arrays by passing null pointers for those arguments which the user does not wish to free.

4. Parameters

a

Input: the non-zeros of the sparse constraint matrix A, to be freed. If **a** or ***a** is a null pointer, no action is taken.

Output: if \mathbf{a} is not null, $*\mathbf{a}$ is set to the null pointer.

ha

Input: the row indices of the non-zero elements stored in \mathbf{a} , to be freed. If \mathbf{ha} or $*\mathbf{ha}$ is a null pointer, no action is taken.

Output: if ha is not null, *ha is set to the null pointer.

ka

Input: the indices indicating the beginning of each column of A, to be freed. If **ka** or $*\mathbf{ka}$ is a null pointer, no action is taken.

Output: if ka is not null, *ka is set to the null pointer.

\mathbf{bl}

Input: the lower bounds of the problem variables and general constraints, to be freed. If **bl** or ***bl** is a null pointer, no action is taken.

Output: if \mathbf{bl} is not null, $*\mathbf{bl}$ is set to the null pointer.

bu

Input: the upper bounds of the problem variables and general constraints, to be freed. If **bu** or $*\mathbf{bu}$ is a null pointer, no action is taken.

Output: if **bu** is not null, ***bu** is set to the null pointer.

xs

Input: a set of initial values for the variables and constraints, to be freed. If xs or *xs is a null pointer no action is taken.

Output: if xs is not null, *xs is set to the null pointer.

5. Error Indications and Warnings

None.

6. Further Comments

In addition to allocating the memory freed by this function, nag_opt_sparse_mps_read (e04mzc) also allocates memory to the **crnames** member of the **options** structure (if the structure is supplied as an argument). The function nag_opt_free (e04xzc) should be used to free this memory. Users should **not** use the standard C function **free(**) for this purpose.

7. See Also

nag_opt_sparse_mps_read (e04mzc) nag_opt_sparse_convex_qp (e04nkc) nag_opt_free (e04xzc)

8. Example

For an example of the use of nag_opt_sparse_mps_free see the documentation for nag_opt_sparse_mps_read (e04mzc).